Tutoria11y: Enhancing Accessible Interactive Tutorial Creation by Blind Audio Producers

Abir Saha
abir@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Thomas B. McHugh*
mchugh@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Anne Marie Piper
ampiper@uci.edu
University of California, Irvine
Irvine, CA, USA

ABSTRACT

Audio production is a skilled practice that requires mastery in highly complex software and hardware tools. Blind audio producers face a steep learning curve where they must learn multiple inaccessible audio production tools in conjunction with workarounds for screen reader support. Learning audio production is made even more challenging due to a scarcity of educational resources geared towards blind people. Grounded in formative interviews and observations with seven blind audio production instructors, we developed Tutoria11y, an extension for GarageBand to support blind audio producers in creating accessible, interactive tutorials that screen reader users can follow to receive step-by-step guidance and confirmation of their actions. Findings from design exploration sessions with five blind instructors highlight how Tutoria11y can support tutorial creation and augment tutorial playback for blind audio producers. We discuss how we can rethink technology's role as a means to amplify, rather than replace, the knowledge of disabled experts.

CCS CONCEPTS

• **Human-centered computing → Empirical studies in accessibility**.

KEYWORDS

accessibility, audio production, blind, vision impairment, screen reader, interactive tutorial

ACM Reference Format:
Abir Saha, Thomas B. McHugh, and Anne Marie Piper. 2023. Tutoria11y: Enhancing Accessible Interactive Tutorial Creation by Blind Audio Producers. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3544548.3580698

1 INTRODUCTION

With the proliferation of technologies in every aspect of modern life, digital technologies have become an integral part of multimedia content creation. Myriad computer-based applications are being developed to support skilled practices of creative work, such as graphics and 3D model design (Illustrator, AutoCAD, SolidWorks), photo editing (Photoshop, Lightroom), video editing (Final Cut Pro, Premier Pro, DaVinci Resolve), digital drawing and painting (Procreate, Fresco), and so on. Audio production is one such form of computer-supported skilled practice which involves turning unedited audio tracks into professional-sounding content by taking them through time-consuming and complex processing workflows named editing, mixing, and mastering.

Becoming proficient in audio production requires understanding how to work with the medium of audio as well as various software tools that support audio production, such as Digital Audio Workstations or DAWs (e.g., Pro Tools, REAPER, Logic Pro, and GarageBand) and effect plugins (e.g., equalizer or reverb). These audio production tools incorporate complex graphical user interfaces that are heavily geared towards sighted users and often lack accessibility support [21, 30, 35]. As part of learning to use these complex tools, blind audio producers must figure out how to coordinate between screen reader software (e.g., VoiceOver, JAWS, NVDA), additional third party accessibility scripts (e.g., Flo Tools, OSARA), and hardware tools to make DAW features accessible [35]. What's more, this steep learning curve is further exacerbated by a lack of accessible learning resources (e.g., tutorials, guides, and documentation) geared towards blind audio producers. Although many online audio and video tutorials exist to help people learn to use audio production tools, these are largely visual in nature and rooted in a sighted instructor's experience with the tools, which can be dramatically different from that of a screen reader user.

In this paper, we focus on understanding and designing to support the creation of screen reader accessible learning resources for audio production tasks. Our work is grounded in interviews and observations with seven blind audio production experts who create their own written guides and audio-video tutorials as well as offer real-time training sessions to support screen reader users in learning audio production tasks in DAWs. Our formative work reveals that blind trainers create screen reader-centric learning resources to reduce challenges associated with widely available audio production tutorials made by sighted people as well as facilitate structured and hands-on learning for novice blind learners, all while managing a complex workflow for recording and editing accessible tutorials. Drawing on these insights, we developed *Tutoria11y*, a macOS based extension for GarageBand to support blind audio producers in creating accessible, interactive tutorials for teaching audio production tasks. Tutoria11y enables screen reader users to quickly create step-by-step instructions and specify actions required to perform custom audio production tasks. Once a tutorial is created, other screen reader users can access the interactive tutorial in GarageBand and receive step-by-step guidance and confirmation

[*] McHugh contributed to this study during his time at Northwestern University.

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3580698

of their actions. We report results from exploratory design evaluation sessions with five blind audio production experts, which detail how participants used the system to generate interactive tutorials as well as how they reacted to the interactive playback experience.

Our work makes three contributions to HCI and accessible computing. First, we contribute new empirical evidence of the complexities screen reader users encounter in learning audio production practices and generating accessible learning resources, extending prior work that highlights how blind screen reader users engage in audio production and music composition tasks [29, 30, 35]. Second, we introduce new techniques to support blind screen reader users in creating and consuming interactive tutorials for audio production tasks. Our design exploration reveals new insights about ways to scaffold accessible learning and training practices among blind audio producers, complementing prior work that focuses on how sighted people can generate interactive guides for screen reader users [34] and other sighted users in various forms of computer-supported skilled work [4, 5, 10, 11, 19, 24, 39]. Third, we synthesize our findings across the two studies to highlight how we might rethink the role of technologies as a means to amplify, rather than replace, disabled experts' knowledge in improving accessibility in computer-supported skilled work practices.

2 RELATED WORK

Our present paper builds upon prior research on accessibility in audio production tools and practices as well as scholarship on interactive guided tutorials.

2.1 Accessibility in Audio Production Tools and Practices

Although there is extensive prior work on the role of digital technologies in audio production [1, 6, 14, 38] and the design of novel interfaces to enhance audio production workflows such as editing [23] and mixing [9, 31], this research largely focuses on sighted people. Research on accessibility of audio production for blind people is still nascent. For example, Metatla and colleagues [21, 22] used sonification techniques to create accessible representations of peak level meter and automation line anchor points. Other researchers created tangible representations of audio waveform for blind people using haptic feedback [16, 37]. Payne et al. [29] developed SoundCells, a browser-based system to support blind musicians in composing music notation with screen readers and generating output in audio, regular print, and braille formats. Prior research also explored various non-conventional mediums such as tabletop objects [26] and gamepads [15] to create novel tools for accessible music and audio production. While this body of research contributes to the design of new accessible software and hardware for audio content creation, there remains a gap in our understanding of accessibility in mainstream audio production tools and practices. One exception is work by Saha and Piper [35], which details how blind professionals piece together accessible and efficient workflows through a combination of mainstream and custom audio production tools and create and maintain accessible learning resources through community efforts. Our work contributes to this prior literature by further understanding and designing to support accessibility in learning audio production tasks.

2.2 Interactive Guided Tutorials

Interactive guided tutorials have garnered much attention within the HCI research community. In contrast to static text and image-based documentation and audio-video tutorials that require people to switch between the tutorials and the application of interest for learning to accomplish a task, interactive tutorials allow people to stay in the context of the application and execute steps directly from the tutorial while receiving contextual assistance and step-by-step instructions to follow along [10, 27, 39]. One of the most prominent approaches for generating interactive tutorials involve tracing and analyzing user interactions on the application. Over the years, researchers have developed a range of systems that automatically generate step-by-step interactive tutorials from user demonstrations to support learning graphical tasks like image editing and graphics design (e.g., Chronicle [12], Toolclips [11], MixT [5], PPTutorial [19], and more [17, 18]). As an example, Pause-and-Play [32] synchronizes the playback of tutorials to a learner's progress by automated pausing and resuming, thereby eliminating the need for the learner to actively control playback (e.g., pause, fast forward, or rewind) for keeping up with the instructions provided in the tutorial while following along. Others built systems to generate mixed-media tutorials from user demonstrations that would retain the benefits of both text and video formats [5, 24] across various applications and platforms [24, 39]. What's more, this prior work shows that learners complete tasks more effectively by interacting with the software through direct manipulation of the tutorial video than with conventional video tutorials [10–12, 25] and that they find interactive tutorials easy to follow, understand, and remember compared to static or video tutorials [39].

2.3 Accessible Interactive Tutorials

Despite this extensive research, prior work on interactive tutorials often involves graphics-heavy interfaces and rarely focuses on accessibility issues or the experiences of disabled content creators and learners. A notable exception is the study by Rodrigues et al. [34], where the researchers built a system through which blind users can learn to perform smartphone-based tasks following interactive playthrough created by sighted people. This work found that contextual task-assistance improved self-efficacy among blind users regarding performing unfamiliar tasks on smartphones (e.g., adding a new contact and sharing a video to Facebook, etc.) and promoted task-based learning. Another study found that a text-entry tutorial that detects errors and suggests corrections in-context increased typing speed and minimized typing errors on smartphones among older adults [13]. While these recent studies on making accessible interactive tutorials are a promising step, they mostly rely on non-expert blind or sighted people [33, 34] or machine learning algorithms [13] for creating interactive tutorials on basic smartphone navigation and typing tasks. Findings from prior work showed various mismatches between information provided by blind and sighted tutorial creators with limited knowledge of accessible tutorial making and information required by blind tutorial users, which prevented effective learning for users [33]. Our work extends this prior literature by understanding and introducing a new system to support expert blind trainers in producing interactive non-visual tutorials for teaching sophisticated audio production tasks to other screen reader users.

3 FORMATIVE STUDY: METHOD

To understand accessibility of audio production learning resources for screen reader users, we conducted interviews and observations with seven blind and visually impaired audio production experts who have experience with offering real-time training or creating tutorials and written guides for blind audio producers. This study was approved by the Institutional Review Board of our university.

3.1 Participants

We recruited seven participants, all of whom identified as men[1] and were advanced or expert users of two or more screen readers (JAWS, NVDA, VoiceOver, Narrator, TalkBack). Four participants were residents of the United States at the time of the research, while three lived in Europe. Most participants created audio/video tutorials and written guides, and some also offered one-on-one or group training, either on their own or through some institute. Participants shared their audio/video tutorials on their own websites or YouTube channels, online communities or blogs they administered, and WhatsApp groups or mailing lists of blind audio producers they were members of. See Table 1 for participants' self-reported visual disability, type of audio production training they offered, and the Digital Audio Workstations (DAWs) they used.

3.2 Procedure

The sessions were conducted by the first author over Zoom between March and May of 2021. We started each session by collecting verbal consent from the participant or by using a GDPR-compliant online form for those residing in EEA countries. During the sessions, we first asked questions about participants' instruction style, what kinds and formats of learning materials they generated, and their rationale behind preferring certain formats of learning materials over others, probing for challenges they encountered and strategies they developed to make tutorials accessible and beneficial for blind learners. Next, we remotely observed participants as they prepared an audio/video tutorial on an audio production task of their own choice. We asked participants to share their screen via Zoom (including computer sound). Participants used their preferred DAWs and screen readers to prepare the tutorials: Josh, Neil, and Owen used REAPER with NVDA screen reader; Phil and Rob used Logic Pro with VoiceOver screen reader; and Leo and Dylan used Pro Tools with VoiceOver. We paid attention to and asked for explanations on how participants performed various steps in their tutorial creation workflow—starting from setting up different software and hardware tools, narrating and enacting task steps during the recording phase, and editing and publishing the recorded tutorials. We ended the sessions with follow-up questions based on our observation of their tutorial recording and/or editing process, such as particular actions made within the recorded tutorials, presentation styles, etc. Each session lasted approximately 90 minutes. Participants received a US$60 Amazon Gift Card for their time and effort. All sessions were recorded and transcribed for analysis.

3.3 Data Analysis

We followed a thematic analysis approach [3] for data analysis. The initial coding of data was primarily led by the first author who is sighted, has experience with multiple DAWs and screen readers, and has been conducting research on accessible audio production for four years. The first author performed open coding on the transcripts focusing on the different forms of audio production training participants offered, their unique experiences with learning audio production in the past and how these experiences informed their current instruction style, and their software and hardware setups for tutorial recording. We periodically streamlined the open codes across different transcripts by merging similar or overlapping codes. The first and third authors regularly met to discuss these codes and refine them to resolve any disagreements, whereupon the first author grouped relevant codes to create a smaller set of axial codes. Through this iterative refinement process, we reworked the codes into three distinct themes that capture core aspects of creating accessible learning resources for screen reader users.

4 FORMATIVE STUDY: FINDINGS

Our analysis revealed three important aspects of how blind audio production trainers create accessible learning resources for new learners with vision impairments. Below we detail the ways in which participants curate their workflow to support the unique needs of screen reader users in understanding audio production tasks and facilitate structured and hands-on learning among novice learners as well as how they manage the complex procedure of pre-processing, recording, and editing tutorials.

4.1 Supporting Screen Reader-Centric Understanding of Audio Production Tasks

Widely adopted audio production tools provide detailed official guides, documentation, and video tutorials[2] to help users get started with these complex tools. In addition, there is also an extensive and growing number of user-generated video tutorials for audio production tasks on YouTube and other social media platforms. However, our formative interviews revealed that much of these resources are geared towards sighted people and do not always align with how screen reader users interact with audio production software. Rob explained that the primary challenge associated with these tutorials *"comes down to the lack of descriptions,"* where sighted tutorial creators do not mention specific names and types of GUI elements and their relative position with respect to nearby elements – all critical information for screen reader users. Additional challenges stemmed from *"the language that gets used around driving software with a mouse"* (Owen) and ambiguous visual-spatial deictic references such as *"I got this plugin up here, so I'm gonna come down here"* (Rob). In addition, a big hurdle in following verbal description of elements in these tutorials is the mismatch between how an

[1] Our all male identifying sample is likely a result of the lack of gender diversity within audio industry. Between 2004 and 2015, only between 8.4% and 15.6% of audio engineers identified as women, averaging around 9% [20]. According to a 2019 study, the percentage of audio producers identifying as women was estimated to be 2.1% [36].

[2] https://www.avid.com/pro-tools/getting-started
https://resources.avid.com/SupportFiles/PT/Pro_Tools_Reference_Guide_2022.12.pdf
https://www.apple.com/logic-pro/resources/
https://www.reaper.fm/videos.php

Table 1: Details of participants. All names are pseudonyms. Participants took part in the formative study (F) and/or design exploration sessions (DE) in Section 6.

Name and Phase	Self-Reported Visual Disability	Type of Training Offered	DAWs Used
Dylan (F, DE)	Totally blind	1:1 training, occasionally audio and written tutorials	Pro Tools (main), REAPER, GarageBand
Josh (F)	Totally blind	Audio (and some video) tutorials, 1:1 and group training	REAPER
Leo (F)	Totally blind	University courses on Pro Tools and audio tutorials on Logic Pro	Pro Tools, Logic Pro
Max (DE)	Some light perception	Audio tutorials	REAPER (main), Pro Tools, and GarageBand
Neil (F)	Totally blind	Audio tutorials and 1:1 training	REAPER
Owen (F)	Totally blind	Audio and written tutorials, 1:1 training	REAPER (main) and Pro Tools
Phil (F, DE)	Totally blind	Audio/video and written tutorials, 1:1 training	Logic Pro
Rob (F, DE)	Some light perception	Audio/video and written tutorials, 1:1 training	Logic Pro (main), REAPER, GarageBand
Seth (DE)	Totally blind, Retinitis Pigmentosa	Written guides	Logic Pro (main) and GarageBand

element appears visually to sighted tutorial makers versus how screen readers describe that element. Rob shared an example of this: *"What is described [by a sighted tutorial creator] as a dropdown menu, VoiceOver calls it a popup button. So some [blind] people are gonna hear 'dropdown menu' [in a tutorial], like 'what's that? I don't see any dropdown menus in here.'"*

The above excerpts highlight the different ways in which tutorials geared towards sighted people fall short in supporting blind audio producers. Recognizing these shortcomings, our participants created screen reader-centric audio production tutorials to reduce *"this gap of figuring out accessible equivalents for the workflows that those [mainstream] tutorials are demonstrating"* (Owen). Our participants drew upon their prior experience, personal workflows, and workarounds developed over time to narrate their tutorials in a way that accounts for the limitations present in visually-oriented tutorials, for example, by referring to GUI elements using the standard terminologies used by screen readers, or by teaching screen reader-centric GUI navigation that does not rely on visual, mouse-based actions such as drag-and-drop. Beyond that, these blind trainers also highlight in their tutorials the inconsistencies and eccentricities associated with interacting with DAWs using screen readers. For example, screen readers may describe UI elements inaccurately or fail to announce certain UI changes due to lack of screen reader support. Such a case of screen reader failing to provide feedback appeared while Phil was recording a tutorial during our session, and he narrated this lack of feedback in his tutorial: *"I'm pressing it (a keystroke) now. VoiceOver will not say anything."* Phil did this so that potential learners would know that this is not due to an error on the learners' part. As another example, Phil clarified in his recorded tutorial how an element was described incorrectly by his screen reader: *"Now this track is called [by the screen reader] 'Komplete'. It's actually a lie. It's 'electric piano.' So ignore that."*

Beyond carefully attending to screen reader feedback in their tutorials, our participants also shared time-saving workarounds and strategies using screen reader navigation and keyboard shortcuts to accomplish otherwise lengthy or complex tasks. Such strategies stemming from their long experience with these tools can uniquely boost productivity for blind learners and are not commonly shared in tutorials made by sighted trainers who are not familiar with the unique challenges of screen reader navigation. During Phil's session, we saw instances of him incorporating such experiential knowledge about screen reader use in the tutorial he recorded:

> ***Phil:*** *We'll press N for Native Instruments because I have far too many [instruments on the list] and it will take years if I don't.*
> ***Screen Reader:*** *Native instruments, sub menu.*
> ***Phil:*** *First letter navigation is a good rule to know because it makes your life a lot easier and speeds up the world.*

Hence, our findings not only shed light on strategies for making audio production tutorials accessible for screen reader users but also underscore the importance of blind trainers' experiential knowledge. Rob commented, *"I'm a screen reader user myself, and I know what I'd wish to see in non-screen reader content (tutorials). So some of it is just innate to me because I'm using these tools in the manner that the people who are watching the content will hopefully want to use it."* Thus, their personal experience with learning and using these tools gives them first-hand insight into the challenges blind learners face and the kind of instructions from which they would benefit. This led us to our **first design goal: centering the experiential knowledge of expert blind trainers.**

4.2 Facilitating Hands-On Structured Learning

In addition to creating tutorial content that focuses on screen reader users, participants also structure their curriculum and education style in ways that resonate with blind learners. One of the primary considerations participants mentioned is to facilitate *"auditory learning"* (Phil) where *"you're listening to someone performing an action or a group of actions"* (Phil). Listening to how a blind trainer performs a task reveals how their screen reader responds to the trainer's different actions and how these actions change the sound or music being produced. Through this, the learner is *"going to be able to vicariously have that experience initially"* (Josh) before actually performing the task themselves. In addition to auditory learning, participants described supporting hands-on practice where the trainers are *"telling you what keys to press, and you can just pause the video or podcast and do those things"* (Phil). Dylan explained that the effectiveness of hands-on practice extends beyond audio tutorials and applies to one-on-one training sessions as well.

> *"When I teach, they (students) are the ones piloting VoiceOver, I'm not. So I'm literally listening to their VoiceOver as they're going through Pro Tools, and I'm telling them what they're hearing — so that it's not just a one-on-one thing where I'm doing all the talking and navigating of the computer, and they're just listening. They're actively engaged... it's like, they're learning from themselves."*

When actions are described and executed by the trainer in rapid succession in audio/video tutorials, it can be overwhelming for new learners to follow along. Participants described trying to alleviate *"that sense of being overwhelmed"* (Owen) among new learners by regulating the pace of instructions, both in tutorials and one-on-one classes, to allow learners to *"go at their own pace"* (Rob). Furthermore, participants described *"breaking stuff (tasks) down into its simplest form"* (Owen) while narrating a tutorial using *"clear, concise directions on what they're looking for, what they need to click on, what they need to navigate to... I try to be extremely specific as to what I'm saying"* (Rob). Dylan explained that some learners who are new computer users may *"have to have everything written out exactly in a list."* In such situations, written guides were helpful. Dylan said he *"needed to write bulleted lists of directions on how to do something. Step one… step two… press this, then this."*

In summary, expert trainers decomposed tasks into small, manageable steps and put them in a format that learners can follow on their own, one step at a time. This led to our **second design goal: scaffolding hands-on guided practice for learners.**

4.3 Managing a Complex Tutorial Recording and Editing Workflow

Our observational and interview data revealed blind audio trainers' complex workflows for creating audio/video tutorials, which involve performing required setup and pre-processing steps, managing a number of tools to execute the recording tasks, and editing and post-processing recorded tutorials. Participants shared that they needed to juggle between a number of additional applications (e.g., BlackHole, Loopback, etc.) to make sure that their recording captures multiple audio streams including audio tracks on DAW, screen reader feedback, and their own narration. Not only do they have to capture these audio streams but they also need to make sure that there is no auditory overlap and the levels of various audio sources are discernible and understandable (e.g., by slowing down screen reader speech rate when creating a tutorial). In some cases, participants prepared a detailed script to follow and practiced the content of a tutorial several times to minimize potential errors during recording. Others try to *"wing it"* in an impromptu manner, as Rob explained. In either case, participants expend substantial time and effort editing the recorded tutorials to get the *"bad bit out"* to reduce any potential for confusion and *"help make this thing more palatable but [also] more educational and informational"* (Rob). For some, editing tutorials (particularly video content) is so difficult and time consuming that it is easier to re-record the entire tutorial. Phil said, *"When I do my YouTube videos, I don't edit because I don't have the capability of taking out bits or adding bits in later. So if it's not right the first time, then I have to do it all over again."*

Although these blind trainers are motivated to create accessible tutorials, doing so means mastering elaborate tools, managing complex recording workflows, and putting in time to edit content so that the tutorials are instructive and appealing. This led to our **third design goal: streamlining the workflow for recording accessible tutorials.**

5 SYSTEM DESIGN AND DEVELOPMENT

To address the three design goals we identified through our formative work, we developed *Tutoria11y*, a macOS application for recording and playing interactive tutorials in GarageBand.

5.1 Description of an Interactive Tutorial

Tutoria11y has two primary modes of use: recording custom interactive tutorials and playing the interactive tutorials. An interactive tutorial created using Tutoria11y contains voice instructions on how to perform a task on GarageBand, much like a regular audio tutorial. However, each interactive tutorial is divided into multiple sections or *steps*. When an interactive tutorial is experienced, it will first play the instructions associated with the first step, and it will wait for a learner to perform the actions described in the instructions. Once the learner performs these actions successfully, the tutorial automatically unpauses itself and plays the instructions for the next step. We define these points where a tutorial stays paused between two consecutive steps waiting for the learner to perform some actions as *breakpoints*.

Each interactive tutorial consists of two files: a *.tutorial* file and a companion GarageBand project file. The GarageBand project file reflects the starting state of the task, and learners will perform the actions described in the tutorial using this project file. As an example, if an interactive tutorial involves unmuting a track, the companion GarageBand project may include a single audio track that is muted. The .tutorial file contains the auditory instructions recorded by the trainer, breakpoint timestamps, and a list of actions needed on the learner's part to complete each of the steps. In the previous example, the list of actions would include a single action — unchecking the mute button on the track.

5.2 Recording Experience

Figure 1 shows the different stages of the recording process. The recording process can be started in two ways – either by clicking on the 'record' button on Tutoria11y's user interface or by pressing a global keyboard shortcut (command-control-R). The second option allows a user to start recording from within GarageBand without having to switch back and forth between Tutoria11y and GarageBand. Once the recording has started (stage 1), the trainer will narrate one step of the task first (stage 2) and then perform

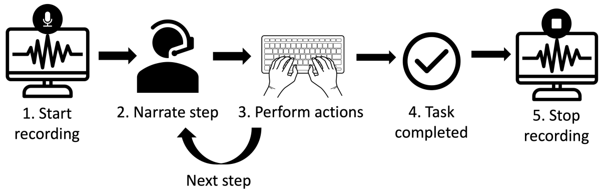

Figure 1: The five stages of the tutorial recording process on Tutoria11y include (1) start recording, (2) narrate step, (3) perform actions associated with a step, (4) repeat stages 2 and 3 until all steps have been narrated and performed, and (5) stop recording.

the actions associated with that particular step themselves (stage 3). Stages 2 and 3 will be repeated for each subsequent step of the task, until all the steps have been narrated and performed (stage 4). When the trainer wants to stop recording (stage 5), they can either press the same keyboard shortcut (command-control-R) from within GarageBand or switch to the Tutoria11y application to click on the 'stop recording' button. A 'save file' dialog box will appear, allowing the trainer to type in a name for the tutorial and save it.

5.3 Playback Experience

Figure 2 shows the different stages of the playback experience. To play an interactive tutorial, a learner first needs to open the companion GarageBand project. Then they will click the 'choose file' button on Tutoria11y's user interface. A 'open file' dialog box appears, and the learner will need to choose a tutorial file from their computer storage, press the 'open' button, and switch back to the GarageBand window. Once the 'open' button has been clicked and the tutorial starts playing (stage 1), the instructions for the first step of the task will be played (stage 2) and playback will pause automatically (stage 3). Once the playback has paused, the user will need to perform the actions associated with this step on the companion GarageBand project (stage 4). Only when the user has successfully completed the actions associated with the first step, Tutoria11y will resume playback and play the instructions for the second step. Stages 2-4 will repeat for each subsequent step, until the learner has successfully performed all the steps and completed the tutorial (stage 5).

5.4 Implementation Details

Tutoria11y is built using Objective-C and Swift. Tutoria11y's user interface contains three elements: a 'start recording/stop recording' toggle button, a 'choose file' button to select and play a tutorial file, and a dropdown menu containing a list of DAWs. Using the dropdown menu, a user can choose which DAW the tutorial they are recording is intended for. The current version of Tutoria11y only supports GarageBand, since we intended to start with a free and basic DAW that comes included with macOS and has decent accessibility support that we could leverage for our system design.

The tutorial recording process requires access to three permissions from macOS: microphone access to record the trainer's voice, macOS accessibility API access to keep track of changes made to the different UI elements within GarageBand, and speech recognition access to detect if the tutorial creator is speaking or not. When a trainer starts the recording process, Tutoria11y first takes a 'snapshot' of the accessibility hierarchy of GarageBand, which essentially saves the state or value of GarageBand UI elements at the beginning of recording. Whenever the trainer performs an action on GarageBand using their keyboard during the recording phase, Tutoria11y takes another snapshot of GarageBand's accessibility hierarchy reflecting the most recent changes made to the UI. Each time a new snapshot is captured, Tutoria11y compares it with the previous snapshot to determine which GUI elements have been manipulated by the trainer between the previous and current snapshots and maintains a list of these UI changes and their timestamps. After the trainer completes recording the tutorial by pressing the 'stop recording' button or shortcut, Tutoria11y applies speech recognition to the trainer's recorded voice to determine the timestamps of silences in-between the trainer's narration. Furthermore, Tutoria11y also checks the list of UI element changes and their timestamps to determine if the tutorial creator performed any actions in the middle of a silence. If Tutoria11y finds any changes made to the GarageBand UI in the middle of a silence, Tutoria11y marks it as a breakpoint in the interactive tutorial. The silent regions that are associated with breakpoints are automatically trimmed out. Overall, the saved tutorial file contains the voice recording of the trainer, the timestamps of the implemented breakpoints, and the list of UI changes associated with each breakpoint.

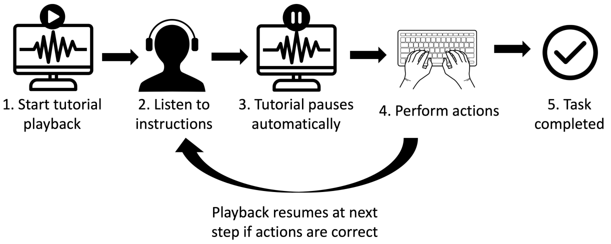
1. Start tutorial playback
2. Listen to instructions
3. Tutorial pauses automatically
4. Perform actions
5. Task completed
Playback resumes at next step if actions are correct

Figure 2: The five stages of tutorial playback experience on Tutoria11y include (1) start playback, (2) listen to instructions for a step, (3) automatic pausing of tutorial, (4) perform the actions associated with the step to make the tutorial automatically unpause, and (5) repeat stages 2-4 for each subsequent step until the tutorial is completed.

When an interactive tutorial is opened for playback, the tutorial will automatically pause at each breakpoint and wait for the learner to replicate the exact UI changes associated with this breakpoint. At each breakpoint, whenever the learner performs an action on GarageBand using their keyboard, Tutoria11y takes a snapshot of GarageBand's accessibility hierarchy and checks if the necessary GUI changes for the current breakpoint have been performed. After successful completion of a breakpoint, playback resumes immediately and Tutoria11y plays the next set of instructions without any silence in-between, since the silent portions associated with breakpoints are automatically trimmed out at the end of the recording process, thus allowing for a seamless playback experience.

Important to acknowledge here is that the current version of Tutoria11y does not account for situations where the learner makes a mistake during playback or the trainer performs an incorrect action during the recording phase, and the playback or recording will need to be restarted from beginning in such situations. Implementing easier ways to rectify such mistakes is an important next step — and our participants also reflect on this in our Findings section.

6 DESIGN EXPLORATION: METHOD

We conducted exploratory evaluation sessions with five blind audio production trainers whose specializations ranged from offering real-time training to creating audio/video tutorials or written guides. Since Tutoria11y was our participants' first time experiencing interactive tutorials of any kind, an exploratory evaluation approach allowed us to observe how each of them recorded their first interactive tutorials based on their own instruction styles. In addition, it allowed them to freely ask us questions and share feedback in real time as they participated in the recording and playback activities. Our overarching goal was to solicit feedback on the recording and playback experiences of Tutoria11y and learn how they envisioned using Tutoria11y and interactive tutorials in their own training process. Our study was approved by the Institutional Review Board at our university.

6.1 Participants

Participants were recruited from our research network and snowball sampling (aged 45-70, all identified as male). Four participants were residents of the United States at the time of this study. Three of them also took part in our formative interviews. For audio production software, all participants used both Logic and GarageBand, although Dylan and Max primarily used Pro Tools. Phil, Rob, and Dylan frequently created audio tutorials and offered professional training to blind learners pursuing audio production. Seth prepared text-based tutorials and written guides for GarageBand, although he did not record audio tutorials. Max did not prepare tutorials on a formal basis, although he provided expert suggestions (in both written and audio format) on online forums for blind audio producers. All participants used VoiceOver as their primary screen reader, although Max and Dylan were also proficient with JAWS and NVDA. See Table 1 for details of participants' self-reported visual disability, type of training offered, and DAWs used.

6.2 Procedure

The first author conducted design exploration sessions with participants via Zoom between March and June 2022. Each session lasted for approximately 90-120 minutes. The session with Seth was divided into two 90-minutes sessions on the same day due to delays caused by technical difficulties.

We started each session by collecting consent from the participants and walking them through the setup procedure for Tutoria11y. Next, to give them an idea of how an interactive audio tutorial works on Tutoria11y, we asked them to open and play a pre-recorded demo tutorial we created that walked them through unmuting a muted audio track and decreasing the volume level of that track. We asked them to share their initial impressions and thoughts on experiencing the interactive audio tutorial.

During the session, participants' main tasks were creating two interactive audio tutorials using Tutoria11y on GarageBand. For the first task, all participants recorded a tutorial on a pre-selected topic: demonstrating how to trim out a silent portion from the middle of a track. For the second task, participants could select any topic on their own, but we asked them to choose a basic audio production task that could be completed in 1-3 minutes. After completing the recording process, participants were invited to play back the interactive tutorials they recorded. While participants were recording the tutorials and experiencing the playback, we took notes on their reactions and remarks. See Table 2 for details about the tutorials created by participants.

Prior to conducting our sessions, we sent participants all the necessary files, including the executable file for Tutoria11y, the pre-recorded demo tutorial and its companion GarageBand project, and another GarageBand project file that contained the necessary audio track and initial GarageBand UI state for the first task. We also provided detailed instructions for installing Tutoria11y and required setup steps on GarageBand. All participants except Phil used Tutoria11y to record and play interactive tutorials on GarageBand. Phil did not have GarageBand installed and used Tutoria11y on Logic to record the audio tutorials; however, he could not play his recorded tutorials himself, since the playback functionality was not yet implemented for Logic. Instead, we played back on our end the demo tutorial and a pre-recorded tutorial on the first topic (created by the research team), while Phil listened to how the playback functionality on Tutoria11y worked through Zoom.

All participants successfully completed recording the tutorials for both tasks, and all participants who used GarageBand experienced the demo interactive tutorial on their computers successfully. In addition, Max and Seth experienced their own recorded tutorials for the first task and Dylan experienced his own tutorials for both tasks. Participants sometimes ran into issues while experiencing one or both of their own recorded tutorials due to memory overflow issues or because their configuration of GarageBand UI during recording (e.g., full-screened window) did not match the configuration during playback — a scenario that Tutoria11y did not account for at that time. Participants who did not experience their own tutorial for the first task instead experienced an interactive tutorial on the same topic but created by the research team.

We concluded the sessions with an overall debrief on the entire Tutoria11y system, probing participants for their thoughts on the recording process and playback experience of interactive tutorials, how they might incorporate a tool like Tutoria11y into their training and tutorial building workflow, how interactive tutorials might shape tutorial playback experiences of blind learners, potential use-cases, trade-offs and challenges that might arise with Tutoria11y in comparison with their current work practices, and their suggestions for further improvement. Participants were compensated with a US$60 gift card. All sessions were recorded via audio and screen capture and transcribed for analysis.

6.3 Data analysis

We analyzed our observational data by reviewing the recorded sessions and coding user interaction with the system, including completion time, errors or points of confusion, and topics of tutorials created (see Table 2). We analyzed their comments and reflections on the system following thematic analysis [3]. Our analysis

Table 2: Details of tutorials participants created. 'Time to record' is the time between performing the 'start recording' and 'stop recording' actions on Tutoria11y. For 'no. of actions in a single step', an action is defined as a keyboard command to trigger an UI change or shortcut-based workflow, or to perform navigation (e.g., move the screen reader focus, keyboard focus, or playhead cursor) on GarageBand. We consider consecutive navigational keypresses (e.g., pressing the down arrow three times inside a menu) as a single action.

Task	Topic	Name of participant	Time to record (min:sec)	No. of breakpoints	Avg. duration of instructions in each step (sec)	No. of actions required in a single step
First task	How to trim out an unwanted region from a track	Phil	3:43	20	7.5	1-2
		Rob	2:51	11	11.5	1-2
		Seth	5:03	7	21.7	1-4
		Max	2:23	3	38.3	2-3
		Dylan	1:12	7	5.9	1-3
Second task	How to split the keyboard to play two separate instruments	Phil	4:58	18	12	1-2
	How to cut an audio region from one track and paste the region to another track and change pitch	Rob	3.25	9	10.7	1
	How to pan a track	Seth	5.36	11	12.6	1-2
	How to change the tempo of a GarageBand project	Max	1:12	1	54	3
	How to create a new track in GarageBand	Dylan	0:37	3	6.3	1-3

involved a process of open and selective coding led by the first author, where we initially focused on our participants' reactions to different aspects of the recording and playback experiences on Tutoria11y. The first and third authors met weekly to review initial codes and data together. Next, we analyzed their reflections on how the experience of recording and playing interactive tutorials on Tutoria11y compared to that of regular audio tutorials to identify how Tutoria11y could lower barriers to tutorial creation and scaffold accessible tutorial playback experiences for blind audio producers. Based on iterative refinement and examination of codes, we developed three distinct themes that capture Tutoria11y's potential role in shaping accessible audio production training.

7 DESIGN EXPLORATION: FINDINGS

Below we describe how participants reacted to the recording and playback experiences of Tutoria11y and envisioned the ways in which Tutoria11y can support blind trainers and learners. One of our primary goals was to understand the extent to which Tutoria11y could support the tutorial creation workflow for blind audio production trainers and how trainers thought the system would augment the tutorial playback experience for blind learners. We also wanted to learn about the scope and applicability of interactive tutorials and how the incorporation of interactive tutorials could shape one-on-one and group training dynamics for our participants.

7.1 Supporting Tutorial Creation Workflow

Based on their experience with recording interactive tutorials using Tutoria11y, our participants shared the different ways in which Tutoria11y could simplify the setup and recording process and reduce the time commitment needed for tutorial creation while streamlining the implementation of breakpoints in interactive tutorials.

7.1.1 Recording Custom Interactive Tutorials. All five participants were able to successfully record a new tutorial for the predefined task of trimming an unwanted region from a track. In most cases, participants completed recording on their first try. Seth and Dylan had to restart recording once because Tutoria11y quit unexpectedly. Although the topic was the same, the tutorials created by our participants had noticeable differences — participants took between 1 min 12 sec (Dylan) and 5 min 3 sec (Seth) to complete the recording, and the number of breakpoints ranged from 3 (Max) to 20 (Phil). See Figure 3 for an example of a tutorial with spoken instructions, keyboard input, and breakpoints. This variation in duration and number of breakpoints was a result of participants' unique presentation styles, workflows, and pace. As an example, when recording the tutorial, Phil played back the track being trimmed at the end of each step to demonstrate how his actions altered it, resulting in a longer tutorial with a higher number of breakpoints. On the other hand, Dylan focused on only performing the actions without playing back the track, resulting in a shorter tutorial with fewer breakpoints. In addition, participants sometimes divided an identical sequence of actions into different number of steps, leading to differences in the number of actions and average length of instructions and for each step. For example, Max first navigated to four seconds on the audio timeline and then split the track at four seconds as part of the same step, whereas Dylan broke this sequence of actions down to two different steps. Seth, who specializes in creating written guides and does not prepare audio tutorials, took more time to gather his thoughts and narrate his steps.

While we do not compare the tutorials participants created for their second task because the topics were different, we noticed that some of our observations regarding their personal styles remained consistent across both tutorials (e.g., Phil playing back the audio track at the end of every step). Furthermore, the tasks performed by participants appeared to be of similar complexity and scope as that of the first task in terms of time needed to record, number of breakpoints, and average number of actions required in each step.

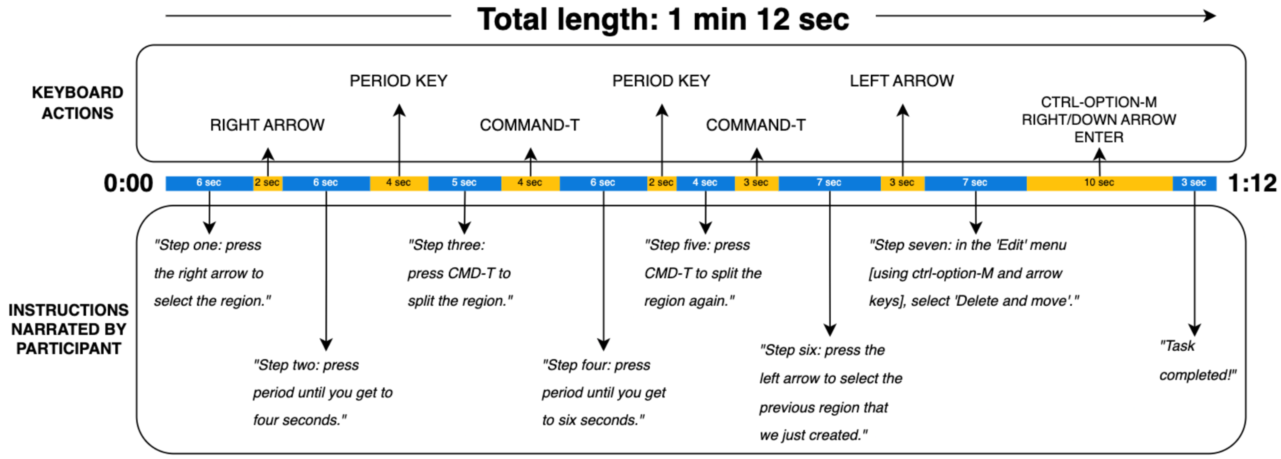
Total length: 1 min 12 sec
KEYBOARD ACTIONS
RIGHT ARROW
PERIOD KEY
COMMAND-T
PERIOD KEY
COMMAND-T
LEFT ARROW
CTRL-OPTION-M RIGHT/DOWN ARROW ENTER
0:00
6 sec
2 sec
6 sec
4 sec
5 sec
4 sec
6 sec
2 sec
4 sec
3 sec
7 sec
3 sec
7 sec
10 sec
3 sec
1:12
INSTRUCTIONS NARRATED BY PARTICIPANT
"Step one: press the right arrow to select the region."
"Step two: press period until you get to four seconds."
"Step three: press CMD-T to split the region."
"Step four: press period until you get to six seconds."
"Step five: press CMD-T to split the region again."
"Step six: press the left arrow to select the previous region that we just created."
"Step seven: in the 'Edit' menu [using ctrl-option-M and arrow keys], select 'Delete and move'."
"Task completed!"

Figure 3: Timeline of the recording process for Dylan's first tutorial. The horizontal bar consisting of blue and yellow chunks represents the time between the start and stop of the recording process (1 min 12 sec). The blue chunks denote the sections where Dylan narrated his instructions. Each yellow time chunk involved Dylan staying silent and performing the actions he had described during the previous blue chunk. Upon completion of recording, Tutoria11y automatically trims out the seven yellow chunks containing silences from the recorded audio and implements one breakpoint in between every adjacent pair of blue chunks (seven in total).

7.1.2 Simplifying Setup and Recording Process. Participants compared their experience with Tutoria11y to their existing tutorial creation process, stating that Tutoria11y is *"easier"* (Max) and *"simple and straightforward"* (Rob). In our formative study, participants had mentioned complexities of their regular tutorial creation workflow, which involved setting up a combination of multiple software applications (for example, Rob mentioned using five different tools – Loopback, Audio Hijack, Soundflower, BlackHole, and QuickTime) to route and capture multiple sources of audio. Upon experiencing Tutorially's recording process in the design exploration study, they went into deeper detail about how complex their existing process was compared to their experience with Tutoria11y. Dylan explained that when recording regular audio tutorials *"I literally have to make sure that Loopback is routed right, whatever recording app I'm gonna use alongside Pro Tools is working right, be inside of Pro Tools and then record."* He contrasted that to Tutoria11y's recording process, which he said *"is just mindlessly easy... It puts me on autopilot almost, because... I'm not thinking about — Oh man, is this recording? Is that mixed? I just need a mic."*

Participants also appreciated that the recording process can be started or stopped using a global keyboard shortcut without leaving the DAW (i.e., GarageBand) or having to switch back and forth between the DAW and the tutorial recording application. Max said, *"That's super handy—to be able to do it in your [DAW] environment and you're not having to fuss with the app (Tutoria11y)."* Dylan added, *"When I actually tell what to do, I'm not concentrating on the technology (Tutoria11y)... I literally just start recording [without leaving GarageBand] and I don't have to go back [to the Tutoria11y window]."* These comments highlight that not having to switch back and forth between multiple applications allowed Tutoria11y to blend into the background and enabled our participants to focus on the content of the tutorials they were recording.

Streamlining the tutorial creation process may also encourage more blind audio experts to create and share their own tutorials. Rob said Tutoria11y has the potential to *"lower the barrier to entry for people that might have knowledge they want to share"* by not only simplifying tutorial creation but also reducing the financial burden of having to buy the expensive software tools mentioned earlier, which currently makes creating tutorials *"a costly proposition"*. Dylan explained, *"Anyone could create this. No one's having to think about 'Well, I don't know how to route JAWS. I don't know how to route VoiceOver.' We're literally saying, 'You know how to use a microphone? Done.'"* Seth exclusively created written documentations and avoided creating audio tutorials due to the complexities and cost of setup and time commitment. After using Tutoria11y, Seth expressed his interest in using the system to create and share tutorials with online communities of GarageBand users. He explained, *"I would love to be able to have a full tutorial out there that people could use, but it's just so much darn work... especially for free, I'm not earning any money on it... And I think to be able to have a program like yours, I could actually even see myself...mak[ing] some tutorials and shar[ing] them with [online forum]."* By simplifying the setup and recording process, Tutoria11y has the potential to broaden the community of creators who are willing to create new training resources and share their knowledge with others.

7.1.3 Editing Interactive Tutorials. Our participants highlighted the amount of editing required as a challenging aspect of their regular tutorial creation process. Even after piecing together multiple necessary applications successfully and recording the tutorial, a trainer needs to go through a time consuming editing process to trim out unwanted portions and silences. Rob explained, *"I try to make all my tutorials so tight and to the point, it does require a lot of editing that is consuming free time that I can't work on other stuff... A 20 min tutorial can be like 2-3 hours of editing."* Rob and Dylan appreciated the automated editing feature of Tutoria11y, which trims silent portions of audio on both sides of a breakpoint and only keeps the relevant portions that contain the instructions for each step. Typically, this editing work needs to be done manually and requires a significant amount of time. Rob said, *"I think the biggest thing will just be time saved... anything like this (Tutoria11y) that gets a lot of the editing out of the way for you is a good thing."*

While participants appreciated the automated trimming of silences in Tutoria11y, they also desired manual controls that would allow them more flexibility in tutorial recording and editing process. In particular, since Tutoria11y's current implementation does not support manual editing of tutorials to rectify mistakes made during the recording process, Max shared that the recording process felt *"strangely stressful... When you're doing something that's complicated and has lots of steps, it almost feels like a lot of pressure. Cause it's like, oh, if I mess up, I have to do the whole thing over again."* To address this, Dylan and Phil wanted the option to re-record certain steps in the event of any mistakes during recording, so that only the steps involving the mistake can be re-recorded without having to record the entire tutorial from the beginning. Phil reflected on an incident that took place when he was recording his first interactive tutorial of the session — he narrated at first that the keyboard focus was on 'track one' although a subsequent screen reader feedback revealed that it was on 'track two', and he later corrected this mistake in his narration. Phil shared that he would like to be able to *"chop out"* the mis-narrated portion of the recording so that only the eventual corrected narration would remain.

7.1.4 Implementing Breakpoints. An important aspect of recording interactive tutorials is implementing breakpoints, i.e., points where the playback of the tutorial will pause automatically and wait for the learner to complete certain actions successfully before resuming playback. During our design exploration sessions, participants created tutorials with varied numbers of breakpoints (see Table 2). For example, in the first task, where each participant had to record a tutorial on the same topic, the number of break points ranged from 3 (Max) to 20 (Phil). Since Tutoria11y is designed to implement these breakpoints without requiring any manual input from the trainer, Dylan appreciated that he could focus solely on narrating and performing the task being demonstrated in the tutorial within the DAW without having to interact with the Tutoria11y app.

We explained to our participants before they started recording the interactive tutorials that breakpoints would be implemented at a point of silence in the recording if Tutoria11y managed to detect any UI changes on GarageBand during that silence. However, predicting exactly what actions or GarageBand UI changes would create a breakpoint remained a source of confusion among our participants. One such incident occurred in the case of the first interactive tutorial recorded by Max. Although Tutoria11y correctly implemented breakpoints for actions such as splitting a track and deleting an unwanted region from a track, Max had also expected another breakpoint to be implemented when he moved his keyboard focus from one UI element to another. Since Tutoria11y's current implementation does not always have access to changes made to the keyboard focus of macOS, there was a mismatch in expectation for Max when he played back the tutorial and found that no breakpoint was implemented for that step (i.e., the playback didn't pause and wait for the learner to switch the keyboard focus before narrating the next set of instructions). To address this, several participants described wanting the ability to manually add (via a keyboard shortcut) and adjust breakpoints after finishing the recording.

7.2 Augmenting Tutorial Playback Experience

As mentioned earlier, all participants who used GarageBand were able to play the demo interactive tutorial on their computer, while three of them were also able to play at least one of their recorded tutorials. Reflecting on their playback experience of interactive tutorials on Tutoria11y, participants shared how interactive tutorials could provide learners with scaffolding for following along instructions, reduce the need for jumping between multiple applications and enable an engaging, gamified tutorial playback experience.

7.2.1 Facilitating Step-by-Step Hands-On Learning. One key goal in designing Tutoria11y was to make it easy for blind learners to follow along with audio tutorials on a step-by-step basis. The trainers in our study emphasized the importance of not only listening to instructions in a tutorial but also *"follow[ing] through with the exact same steps"* (Phil) demonstrated by the trainer. Multiple instructors, however, mentioned that pausing and resuming the tutorial so that one can try the steps on their own is difficult, particularly when the tutorial is complex. In contrast to this experience, participants appreciated how interactive tutorials prepared through Tutoria11y divide large tasks into *"bite-sized"* (Seth) steps by incorporating breakpoints, reducing the need for new learners to figure out when to pause or resume playback of instructions and start replicating them. Rob called the automated generation of small, concise steps a *"game changer"* and *"killer feature"*. He explained, *"From what I've seen in my experience doing one-on-one training, that's the type of thing that is gonna really make a certain group of people that want to learn, feel comfortable learning and being able to really do it at their own pace."* Phil saw Tutoria11y functioning as *"an educational tool"* that provides additional scaffolding to screen reader users as they listen to tutorials. Participants suggested incorporating mechanisms to easily jump back and forth between prior and current steps so that learners can work *"at a comfortable pace"* (Rob) .

Participants pointed out that creating interactive tutorials that have an appropriate number of steps and number of keyboard actions associated with each step would require careful attention from the trainer. Reflecting on the demo tutorial's first step which required the learner to complete six actions and remember the names and locations of four UI elements, Seth wondered *"if a relatively new person who's not that experienced with VoiceOver would've completely followed every one of those steps."* Indeed, our participants mentioned deliberately keeping steps simple and concise when they recorded interactive tutorials in our sessions. Expert tutorial makers Phil, Rob, and Dylan spent less than 9 seconds on average to narrate the instructions for a single step and used between 1 and 4 keystrokes for each instruction (see Table 2), suggesting this might be an ideal scope for beginning learners.

7.2.2 Minimizing the Need to Jump between Multiple Applications. While traditional audio tutorials also have the pause and rewind capabilities, participants spelled out how learners may face difficulties with *"switching back and forth"* between multiple devices, browser tabs, and/or application windows [10–12, 39] to simultaneously keep track of verbal instructions on the tutorial interface and replicating those instructions on their own in the DAW window. This process becomes especially complicated *"for beginners and new users... They might be command-tabbing [to switch between multiple applications] for 5-10 seconds through everything else [to] get to GarageBand"* (Rob). In contrast, interactive tutorials that enable capabilities for pausing and replicating steps within the same application reduces the need to muddle through multiple software and/or hardware applications, enabling blind learners to focus on one smaller task at a time and thus minimize the cognitive effort associated with multitasking. Rob said, *"This just allows you to focus on one task at a time and not have to try and figure out how to go back and forth between a couple of things while trying to learn this one task."* Dylan further explained the challenges with multitasking for neurodivergent screen reader users, adding how interactive tutorials could help them manage attention.

> *"I deal with people who are neurodivergent and I am ADHD... Some folks are ADHD to the point where too many tasks [can make them feel] like... 'I'm behind now... Now I gotta pause. I gotta rewind...' It (interactive tutorial) puts it into like a single tunnel focus of 'just do what it says.'"*

Overall, these findings highlight how the playback of interactive tutorials on Tutoria11y that does not require switching between multiple apps and devices reduces the cognitive load associated with trying to follow along instructions while listening to a tutorial.

7.2.3 Enhancing Tutorial Playback experience through Gamification.

A surprising insight from our analysis was that some participants thought of the interactive tutorials as a gamified approach to learning audio production. They shared that the *"reactionary comments"* from trainers after accomplishing each step could work as a *"natural encouragement mechanism to keep moving forward"* (Rob). Rob explained that if the recorded instructions of an interactive tutorial *"react to the person completing the task — 'Congratulations! All right, now you know how to do this,' 'Excellent! Now the next step'... It makes it a little bit more welcome."* To better support this gamification aspect, participants felt that trainers may need to focus on *"adapting my style to meet an interactive format"* (Rob), including pre-planning tutorial steps and following a script for narration so that they do not forget to add congratulatory messages. Indeed, during our sessions, only Max remembered to put in encouraging remarks in the middle of his tutorials, and all participants except Max forgot to add a congratulatory remark at the end in at least one of their recorded tutorials. Participants thought that *"a reminder that I have to record bookends at the end"* (Dylan) and a *"best practices"* documentation included within the Tutoria11y app would be beneficial for trainers in getting used to making interactive tutorials.

While referring to the gamification aspect as *"groundbreaking"* (Phil), participants put emphasis on reliably and accurately tracking whether a learner has actually completed a step or not. As an example, during playback of the demo tutorial in Phil's session, an unintended keyboard action that was not suggested in the instruction still resulted in the UI change required to complete a step, and Tutoria11y progressed to the next set of instructions. In such cases, Phil and Rob cautioned against incorrectly *"rewarding"* a learner even if they perform an action not suggested in the tutorial or fail to complete a task altogether *"because that can also be very misleading"* (Rob). Max pointed out that the current playback experience on Tutoria11y *"is very oriented around the happy path (a scenario where learners successfully complete all steps without errors). In my experience, tutorials, especially for new blind users, very rarely go on the happy path."* Therefore, Max and Phil suggested allowing trainers to record additional instructions for each breakpoint to provide more guidance in the event of incorrect actions from learners.

Collectively, the thoughts and suggestions our participants shared regarding Tutoria11y's playback experience reveal important insights on how interactive tutorials could support blind learners by providing a streamlined and unobtrusive environment to practice a task by following instructions and engaging them through a gamified experience.

7.3 Scope and Applicability of Accessible Interactive Tutorials

Interactive tutorials have been extensively studied in academic research (e.g., [5, 10, 11, 24, 39]) and commercially available on industry tools (e.g., Adobe Lightroom) to support sighted users. However, blind audio producers who participated in our study had traditionally come to think of interactive tutorials as *"a very visual thing that we (blind users) have not been able to have"* (Dylan). To this end, participants shared their thoughts on the potential use cases of Tutoria11y and interactive tutorials more broadly.

7.3.1 Learning Screen Reader Interaction with GUI Elements.

Participants thought that interactive tutorials for blind learners would be appropriate for concrete, screen reader navigation oriented tasks, but the scope could be *"expanded beyond audio [production]"* (Dylan) to cover basic computer use (e.g. saving a text file) as well as other forms of computer-supported skilled work more broadly. In their opinion, interactive tutorials prepared on Tutoria11y would be beneficial for *"getting started"* (Max) with audio production and *"learning how to use the [DAW] software in an accessible way"* (Rob). Particularly when practicing tasks that involve exceptional forms of screen reader navigation or interaction with complex DAW features, blind learners may miss important details if they are only listening to a regular audio tutorial passively without also performing the interaction simultaneously. To illustrate this point, Phil shared the example of 'inspector table,' a GUI element on Logic that cannot be properly manipulated using keyboard alone and must be controlled by simulating mouse clicks. This is an example of a rather complicated form of interaction where participants felt that interactive tutorials could provide enhanced support for blind learners to comprehensively understand and practice such tasks. In contrast, Rob and Seth expressed concerns about whether interactive tutorials would be suitable for tasks with less structure that require *"more fundamental learning... like improving your mixing...[and] audio skill"* and involve *"artistic creative parts"* where *"it's more a matter of taste and how someone wants something to sound."*

7.3.2 Enhancing One-on-One and Group Training Experiences.

Participants who offered one-on-one or group training to blind audio producers on a professional basis shared ideas about how they would incorporate Tutoria11y and interactive tutorials into their instruction pipeline. Phil shared that he would create interactive tutorials for preparing their DAWs for the tasks that would be taught during class, so that his students can get the preparatory stuff *"ready to go before we actually have the in-person one-to-one lessons."* Dylan envisioned further enhancements made to Tutoria11y that would help him track the progress of his students and allow him to tailor his training materials to suit the needs of individual students. He proposed the implementation of a learner-side log file that would record the timestamps and actions of a student as they experience and complete an interactive tutorial. This log file could allow Dylan to check whether his students completed all the tutorials successfully and understand which steps of the tutorial a student struggled with (e.g., took more time or pressed incorrect keyboard shortcuts). He explained, *"The log could show me what buttons they (students) have pressed that were incorrect. Like they pressed T and that didn't do anything... 'cause they couldn't remember what it was."* According to Dylan, it would make the progress of his students *"calculable, like there's a way for this thing to actually show you metrics."*

In summary, participants saw great potential in using interactive tutorials across tasks with varying complexity and in a variety of learning environments involving one-on-one or group lessons.

8 DISCUSSION

Prior work within HCI and accessibility has called attention to the ways in which blind people must learn, navigate, and maintain a wide range of inaccessible tools to perform different forms of computer-supported skilled work (e.g., [2, 7, 8, 28]). As our study and prior research shows [30, 35], blind audio producers piece together accessible workflows through years of experience with navigating mainstream audio production tools (many of which are inaccessible), leverage their experiential knowledge to create and share unofficial accessibility scripts that boost efficiency of screen reader users, and advocate for improving accessibility in commercial tools with software developers. Moreover, they actively create access for others by passing on their knowledge through written guides, audio/video tutorials, and one-on-one and group teaching and by maintaining question-and-answer forums specifically geared towards blind audio producers. Despite the challenges associated with making accessible tutorials—such as having to buy and learn an extensive suite of software and hardware tools and spending hours to record and edit tutorials without financial remuneration—our participants deeply valued the joy and sense of purpose they received from sharing their knowledge with others. The design of Tutoria11y is meant to augment the existing efforts of blind audio producers and provide an accessible form of scaffolding to learn audio production for screen reader users. Below we discuss three key tensions in designing technologies that support the creation and use of accessible interactive tutorials for audio production and other computer-supported skilled practices more broadly as well as potential areas for improvement and future research.

8.1 Balancing Automation with User Expertise and Control

One key benefit of Tutuoria11y is that it automates tedious components of the tutorial creation workflow (e.g., by automatically trimming out silence and adding breakpoints). By doing so, the system facilitates conditions under which blind experts can focus more on narrating instructions and demonstrating corresponding actions and *"not concentrating on the technology"* (Dylan) for recording the tutorial, which could potentially save time so they can *"get more content out"* (Rob). On the surface, a deterministic view of accessibility may assert automating the entire workflow for generating accessible tutorials as an ideal design goal. However, our work illustrates that technological interventions that automate the entire process of tutorial creation may attempt to replace blind trainers' expertise and experiential knowledge that are an integral component of what makes these learning materials accessible. Indeed, we observed how Phil—while recording the tutorial for an audio production task—explained idiosyncratic behavior of screen readers, alerted listeners when screen readers misrepresented a particular GUI element, detailed efficient ways for screen reader navigation (e.g., first-letter navigation), and provided descriptive instructions for interacting with the complex DAW interface. Without Phil's screen reader-centric instructions, these tutorials would not have provided enough information required by novice blind learners [33] who are just getting started with audio production tasks. Put differently, blind trainers' rich experience with learning and figuring out accessible ways of audio production as screen reader users themselves make them uniquely suited to understand the challenges new learners with vision impairments face and accordingly tailor the learning resources they create for their target audience. While automating the process of tutorial creation based on user logs [5, 10, 12, 18, 39] or inviting sighted authors to generate interactive tutorials for screen reader users [34], as prior work has explored, could be one way forward, our work demonstrates another way of viewing the role of technology in accessible learning. In particular, we argue that for skilled practices like audio production, integrating disabled content creators' knowledge is imperative to ensure accessibility of the learning resources and also honor their professional expertise, advocacy, and community efforts [35].

8.2 Managing Context Switching across Multiple Interfaces

Prior work has introduced a range of systems that generate interactive tutorials to provide contextual assistance, i.e., learners receive guidance in the actual task interface, as opposed to traditional audio-video or text-based tutorials that require learners to repeatedly switch between the task interface and the tutorial window [10, 27, 32, 39]. Our work demonstrates that the challenges with context switching are magnified for blind content creators who need to juggle between not only the task interface (e.g., Digital Audio Workstations or DAWs) and the tutorial playback interface (e.g., a browser window or a separate device for playback) but also manage screen readers as well as additional plugins and scripts (e.g., OSARA, Flo Tools) that are required to navigate inaccessible DAWs. Particularly in the context of audio production, rapidly shifting attention between spoken instructions in tutorials, auditory feedback from screen readers, and various audio tracks and effects in the DAWs can be cognitively overwhelming to screen reader users.

This experience is even more demanding for screen reader users who want to create tutorials for others. Our participants described managing as many as five different application interfaces at a time to make sure that their own voice through microphone, auditory feedback from screen reader, audio tracks on the DAWs, and also occasionally screencast videos or real-time interaction with students are all routed through appropriate channels, have discernible volume and speech rate, and are recorded properly in the resultant tutorials. Our expert participants have honed their skills over the years and are able to maintain this sophisticated workflow, but they highlighted how this complex process prohibits other blind audio producers from fully participating in creating accessible learning resources. While findings from our design exploration with Tutoria11y are promising, particularly the potential for a simplified and streamlined workflow, they point to a larger, systemic problem. Until content production tools become more widely accessible and easy to use with a screen reader, blind content creators must continue to put in extensive time and effort to learn, use, and share their knowledge of these tools with others.

8.3 Supporting Task-Based Learning vs Learning Higher-Level Skills

Prior work details how audio production for screen reader users is as much about learning how to navigate state-of-the-art digital audio tools as it is about learning the craft of audio production [35]. That is, knowing how to use pervasive audio production tools is a crucial part of what it means to be a skilled and proficient audio production engineer. The design of Tutoria11y addresses this challenge by introducing a new way to create step-by-step interactive instructions for blind learners. Following from prior work [27], guided tutorials can foster observational learning that enable novice blind users to learn by (auditorily) observing and replicating actions executed by expert blind audio producers. The aim is for interactive tutorials created through Tutoria11y to further scaffold novice learners by dividing complex instructions into *"bite-sized,"* concise steps, an approach taken in prior work as well [5, 10, 11, 39]. Such incremental, task-oriented learning can be particularly beneficial for new learners in developing self-efficacy, especially when they are just getting started [34]. Yet, the decompositional nature of Tutoria11y may have drawbacks in that it may encourage learners to focus narrowly on executing steps required for a task rather than understanding the process at a higher level. That is, the ability to follow steps in a tutorial does not necessarily mean that users are able to use the tools fluently on their own. Future work must examine the kinds of tasks that are best supported by this approach and how blind instructors envision creating tutorials that teach higher-level skills required for audio production.

8.4 Limitations and Future Work

We acknowledge several limitations in our present paper and possible directions for future research. First, the current version of Tutoria11y does not allow trainers to rectify incorrect actions or mistakes in the narrated instructions without re-recording the entire tutorial. As our participants noted, allowing trainers to manually edit and re-record steps of an existing interactive tutorial will be an important feature to implement in future. Another notable limitation of Tutoria11y's current version is that it does not provide notifications to learners when they take a deviant path, e.g., perform incorrect actions in a step. As such, to enable more effective learning experiences, future iterations need to look into ways to make learners aware of their mistakes and provide opportunities for course correction. This could be done by either pushing error alerts with earcons or spoken notifications from screen readers [34] or playing narrated instructions that trainers may have previously recorded in their own voice for deviant paths. Finally, to gain a deeper understanding of the effects of interactive tutorials on accessible learning of audio production compared to non-interactive tutorials, evaluation of Tutoria11y's playback experience with visually impaired students and beginners is an important future step.

9 CONCLUSION

With an overarching goal of supporting the creation of screen reader accessible learning resources for audio production tasks and grounded in interviews and observations with seven blind trainers, we developed Tutoria11y, an extension that supports blind audio producers in recording and experiencing accessible, interactive tutorials for GarageBand. Our design evaluation sessions with five blind trainers revealed the ways in which Tutoria11y could streamline and simplify accessible tutorial creation, augment and scaffold tutorial playback experiences for screen reader users, and complement real-time training sessions offered by our participants. Synthesis of our findings across both studies encourages rethinking the role of technology in accessible learning as one that supports, rather than automates or replaces, the knowledge of disabled trainers. Furthermore, we encourage future research to investigate how the lessons learned from Tutoria11y's task-based approach could translate into accessible learning resources for the acquisition of higher-level skills among blind learners.

ACKNOWLEDGMENTS

This work was supported by NSF grant IIS-1901456. We thank our participants for their contributions to this study. We are also grateful to Darren Gergle, Marcelo Worsley, Bryan Pardo, and Maitraye Das for their thoughtful suggestions and support at various stages of this work, and to our reviewers for their feedback on earlier drafts.

REFERENCES

[1] Joe Bennett. 2018. Songwriting, Digital Audio Workstations, and the Internet. In *The Oxford Handbook of the Creative Process in Music*, Nicolas Donin (Ed.). Oxford University Press, Oxford. https://doi.org/10.1093/oxfordhb/9780190636197.013.28

[2] Stacy M. Branham and Shaun K. Kane. 2015. The Invisible Work of Accessibility: How Blind Employees Manage Accessibility in Mixed-Ability Workplaces. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility* (Lisbon, Portugal) *(ASSETS '15)*. Association for Computing Machinery, New York, NY, USA, 163–171. https://doi.org/10.1145/2700648.2809864

[3] Virginia Braun and Victoria Clarke. 2006. Using Thematic Analysis in Psychology. *Qualitative Research in Psychology* 3, 2 (Jan. 2006), 77–101. https://doi.org/10.1191/1478088706qp063oa

[4] Minsuk Chang, Anh Truong, Oliver Wang, Maneesh Agrawala, and Juho Kim. 2019. How to Design Voice Based Navigation for How-To Videos. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3290605.3300931

[5] Pei-Yu Chi, Sally Ahn, Amanda Ren, Mira Dontcheva, Wilmot Li, and Björn Hartmann. 2012. MixT: Automatic Generation of Step-by-Step Mixed Media Tutorials. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 93–102. https://doi.org/10.1145/2380116.2380130

[6] Bill Crow. 2006. Musical Creativity and the New Technology. *Music Education Research* 8, 1 (March 2006), 121–130. https://doi.org/10.1080/14613800600581659

[7] Maitraye Das, Darren Gergle, and Anne Marie Piper. 2019. "It Doesn't Win You Friends": Understanding Accessibility in Collaborative Writing for People with Vision Impairments. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (Nov. 2019), 191:1–191:26. https://doi.org/10.1145/3359293

[8] Maitraye Das, Thomas Barlow McHugh, Anne Marie Piper, and Darren Gergle. 2022. Co11ab: Augmenting Accessibility in Synchronous Collaborative Writing for People with Vision Impairments. In *CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, Article 196, 18 pages. https://doi.org/10.1145/3491102.3501918

[9] György Fazekas and Mark Sandler. 2007. Intelligent Editing of Studio Recordings with the Help of Automatic Music Structure Extraction. In *Audio Engineering Society Convention 122*. Audio Engineering Society.

[10] Floraine Grabler, Maneesh Agrawala, Wilmot Li, Mira Dontcheva, and Takeo Igarashi. 2009. Generating Photo Manipulation Tutorials by Demonstration. In *ACM SIGGRAPH 2009 Papers* (New Orleans, Louisiana) *(SIGGRAPH '09)*. Association for Computing Machinery, New York, NY, USA, Article 66, 9 pages. https://doi.org/10.1145/1576246.1531372

[11] Tovi Grossman and George Fitzmaurice. 2010. ToolClips: An Investigation of Contextual Video Assistance for Functionality Understanding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 1515–1524. https://doi.org/10.1145/1753326.1753552

[12] Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2010. Chronicle: Capture, Exploration, and Playback of Document Workflow Histories. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology* (New York, New York, USA) *(UIST '10)*. Association for Computing Machinery, New York, NY, USA, 143–152. https://doi.org/10.1145/1866029.1866054

[13] Toshiyuki Hagiya, Toshiharu Horiuchi, and Tomonori Yazaki. 2016. Typing Tutor: Individualized Tutoring in Text Entry for Older Adults Based on Input Stumble Detection. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 733–744. https://doi.org/10.1145/2858036.2858455

[14] Brian J. Hracs. 2012. A Creative Industry in Transition: The Rise of Digitally Driven Independent Music Production. *Growth and Change* 43, 3 (2012), 442–461. https://doi.org/10.1111/j.1468-2257.2012.00593.x

[15] Björn Kaiser, Matthias Lindner, Sascha Reinhold, and Michael Teistler. 2018. Audio Editing with a Game Controller for Blind Users. In *Mensch und Computer 2018 - Tagungsband*, Raimund Dachselt and Gerhard Weber (Eds.). Gesellschaft für Informatik e.V., Bonn. https://doi.org/10.18420/muc2018-mci-0439

[16] Aaron Karp and Bryan Pardo. 2017. HaptEQ: A Collaborative Tool For Visually Impaired Audio Producers. In *AM'17*. ACM Press, 1–4. https://doi.org/10.1145/3123514.3123531

[17] Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. 2013. Community Enhanced Tutorials: Improving Tutorials with Multiple Demonstrations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1779–1788. https://doi.org/10.1145/2470654.2466235

[18] Ben Lafreniere, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2014. Investigating the Feasibility of Extracting Tool Demonstrations from In-Situ Video Content. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 4007–4016. https://doi.org/10.1145/2556288.2557142

[19] Guo Li, Tun Lu, Jiang Yang, Xiaomu Zhou, Xianghua Ding, and Ning Gu. 2015. Intelligently Creating Contextual Tutorials for GUI Applications. In *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*. 187–196. https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.50

[20] Marlene Mathew, Jennifer Grossman, and Areti Andreopoulou. 2016. Women in Audio: Contributions and Challenges in Music Technology and Production. In *Audio Engineering Society Convention 141*. http://www.aes.org/e-lib/browse.cfm?elib=18477

[21] Oussama Metatla, Nick Bryan-Kinns, Tony Stockman, and Fiore Martin. 2015. Sonifications for digital audio workstations: Reflections on a participatory design approach. In *International Conference on Auditory Display (ICAD)*.

[22] Oussama Metatla, Fiore Martin, Adam Parkinson, Nick Bryan-Kinns, Tony Stockman, and Atau Tanaka. 2016. Audio-Haptic Interfaces for Digital Audio Workstations. *Journal on Multimodal User Interfaces* 10, 3 (Sept. 2016), 247–258. https://doi.org/10.1007/s12193-016-0217-8

[23] Joshua Mycroft, Tony Stockman, and J. D. Reiss. 2018. A Prototype Mixer to Improve Cross-Modal Attention During Audio Mixing. In *Proceedings of the Audio Mostly 2018 on Sound in Immersion and Emotion* (Wrexham, United Kingdom) *(AM'18)*. Association for Computing Machinery, New York, NY, USA, Article 3, 7 pages. https://doi.org/10.1145/3243274.3243290

[24] Alok Mysore and Philip J. Guo. 2017. Torta: Generating Mixed-Media GUI and Command-Line App Tutorials Using Operating-System-Wide Activity Tracing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 703–714. https://doi.org/10.1145/3126594.3126628

[25] Cuong Nguyen and Feng Liu. 2015. Making Software Tutorial Video Responsive. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1565–1568. https://doi.org/10.1145/2702123.2702209

[26] Shotaro Omori and Ikuko Eguchi Yairi. 2013. Collaborative Music Application for Visually Impaired People with Tangible Objects on Table. In *ASSETS'13*. ACM Press, 1–2. https://doi.org/10.1145/2513383.2513403

[27] George Palaigeorgiou and Theofanis Despotakis. 2010. Known and Unknown Weaknesses in Software Animated Demonstrations (Screencasts): A Study in Self-Paced Learning Settings. *Journal of Information Technology Education* 9 (01 2010), 81–98.

[28] Maulishree Pandey, Vaishnav Kameswaran, Hrishikesh V Rao, Sile O'Modhrain, and Steve Oney. 2021. Understanding Accessibility and Collaboration in Programming for People with Visual Impairments. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1, Article 129 (April 2021), 30 pages. https://doi.org/10.1145/3449203

[29] William Payne, Fabiha Ahmed, Michael Gardell, R. Luke DuBois, and Amy Hurst. 2022. SoundCells: Designing a Browser-Based Music Technology for Braille and Print Notation. In *Proceedings of the 19th International Web for All Conference* (Lyon, France) *(W4A '22)*. Association for Computing Machinery, New York, NY, USA, Article 14, 12 pages. https://doi.org/10.1145/3493612.3520462

[30] William Christopher Payne, Alex Yixuan Xu, Fabiha Ahmed, Lisa Ye, and Amy Hurst. 2020. How Blind and Visually Impaired Composers, Producers, and Songwriters Leverage and Adapt Music Technology. In *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (Virtual Event, Greece) *(ASSETS '20)*. Association for Computing Machinery, New York, NY, USA, Article 35, 12 pages. https://doi.org/10.1145/3373625.3417002

[31] Enrique Perez-Gonzalez and Joshua Reiss. 2009. Automatic Equalization of Multichannel Audio Using Cross-Adaptive Methods. In *Audio Engineering Society Convention 127*. Audio Engineering Society.

[32] Suporn Pongnumkul, Mira Dontcheva, Wilmot Li, Jue Wang, Lubomir Bourdev, Shai Avidan, and Michael F. Cohen. 2011. Pause-and-Play: Automatically Linking Screencast Video Tutorials with Applications. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 135–144. https://doi.org/10.1145/2047196.2047213

[33] André Rodrigues, André Santos, Kyle Montague, Hugo Nicolau, and Tiago Guerreiro. 2019. Understanding the Authoring and Playthrough of Nonvisual Smartphone Tutorials. In *Human-Computer Interaction – INTERACT 2019*, David Lamas, Fernando Loizides, Lennart Nacke, Helen Petrie, Marco Winckler, and Panayiotis Zaphiris (Eds.). Springer International Publishing, Cham, 42–62. https://doi.org/10.1007/978-3-030-29381-9_4

[34] André Rodrigues, André R.B. Santos, Kyle Montague, and Tiago Guerreiro. 2021. Promoting Self-Efficacy Through an Effective Human-Powered Nonvisual Smartphone Task Assistant. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW1, Article 114 (apr 2021), 19 pages. https://doi.org/10.1145/3449188

[35] Abir Saha and Anne Marie Piper. 2020. Understanding Audio Production Practices of People with Vision Impairments. In *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility* (Virtual Event, Greece) *(ASSETS '20)*. Association for Computing Machinery, New York, NY, USA, Article 36, 13 pages. https://doi.org/10.1145/3373625.3416993

[36] Stacy L. Smith, Marc Choueiti, Katherine Pieper, Hannah Clark, Ariana Case, and Sylvia Villanueva. 2019. *Inclusion in the Recording Studio? Gender & Race/Ethnicity of Artists, Songwriters, & Producers across 700 Popular Songs from 2012-2018*. Technical Report. USC Annenberg Inclusion Initiative. http://assets.uscannenberg.org/docs/aii-inclusion-recording-studio-2019.pdf

[37] Atau Tanaka and Adam Parkinson. 2016. Haptic Wave: A Cross-Modal Interface for Visually Impaired Audio Producers. In *CHI'16*. ACM Press, 2150–2161. https://doi.org/10.1145/2858036.2858304

[38] Michael Terren. 2019. *The Grain of the Digital Audio Workstation*. Dissertation. Edith Cowan University. https://ro.ecu.edu.au/theses/2201

[39] Cheng-Yao Wang, Wei-Chen Chu, Hou-Ren Chen, Chun-Yen Hsu, and Mike Y. Chen. 2014. EverTutor: Automatically Creating Interactive Guided Tutorials on Smartphones by User Demonstration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 4027–4036. https://doi.org/10.1145/2556288.2557407