

Sonification of Data – A Usability Study

Jacob Kristensson

Division of Rehabilitation Engineering Research | Department of Design
Sciences Faculty of Engineering LTH | Lund University

2023

Master Thesis



Sonification of Data – A Usability Study

Jacob Kristensson
ja0032kr-s@student.lu.se

May 31, 2023

Master thesis work carried out at
the Department of Design Science, Faculty of Engineering, Lund University.

Supervisor: Kirsten Rasmus-Gröhn, kirsten.rasmus-grohn@certec.lth.se

Examiner: Joakim Eriksson, joakim.eriksson@design.lth.se

Sonification of Data – A Usability Study

Copyright © 2023 Jacob Kristensson

Published by

Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O Box 118, SE-221 00 Lund, Sweden

Subject: Interaction Design (MAMM01)
Division: Rehabilitation Engineering Research
Supervisor: Kirsten Rasmus-Gröhn
Examiner: Joakim Eriksson

Abstract

Sonification is a technique that allows people with visual impairment to listen to information that is normally available visually such as in graphs. Not all information is available for people with visual impairment and sonification is not a standard in all places where information is showed, so developing a sonification tool that provides an intuitive experience with easily interpreted data would help a lot of people. Specifically the thesis aims to answer the following research questions: 1. How should auditory characteristics be configured to enable smooth interpretation of sonified data? 2. How do you design a software prototype for someone with visual impairment? 3. In what ways can the end user interact with the sonified data through the prototype?

Through a user-centered design process looking at existing sonification tools, performing usability tests and receiving user feedback a prototype was developed, the research questions were answered, and potential future work was identified.

The findings indicate that correct usage of a screen-reader, providing context through reference values, using big ranges for the auditory characteristics of pitch and note velocity, and users being able to modify the note duration all help in interpreting sonified data. Sonified data can be interacted with by starting, stopping, choosing position, volume, and navigating with a screen-reader. The main guidelines in developing a prototype for people with visual impairment are: Keep the prototype minimalistic, enable keyboard navigation and interaction, and make it screen-reader compatible.

Keywords: sonification, mappings, auditory characteristics, usability, visual impairment, screen-reader

Sammanfattning

Sonifiering är ett verktyg som tillåter personer med synnedsättning att lyssna på information som annars vanligtvis finns tillgängligt visuellt som i till exempel grafer. All information finns inte tillgängligt för personer med synnedsättning och sonifiering är inte en standard på alla ställen som visar info, så att utveckla ett sonifierings-verktyg som ger en intuitiv upplevelse med data som är enkel att förstå skulle hjälpa många människor. Specifikt så vill den här avhandlingen svara på följande forskningsfrågor: 1. Hur ska auditiva egenskaper konfigureras för att möjliggöra en smidig tydning av sonifierad data? 2. Hur designar man en mjukvaru-prototyp för någon som har synnedsättning? 3. På vilka sätt kan slutanvändaren interagera med sonifierad data genom prototypen?

Genom en användar-centrerad design process så tittades det på existerande sonifieringsverktyg, användbarhetstester utfördes och med återkoppling från användarna så ledde det till att en prototyp utvecklades, att forskningsfrågorna blev besvarade och att potentiellt framtida arbete upptäcktes.

Resultaten indikerar att rätt användande av skärmläsare, sammanhang som förses genom referensvärden, användning av stora intervall för de auditiva egenskaperna tonhöjd och nothastighet, samt att användarna kan modifiera notlängden alla hjälper med tydningen av sonifierad data. Sonifierad data kan interageras med genom att starta, stoppa, välja position, volym och genom skärmläsarnavigering. De viktigaste riktlinjerna när man utvecklar en prototyp för personer med synnedsättning är att: Hålla prototypen minimalistisk, göra det möjligt att navigera och interagera med prototypen genom tangentbordet och att göra prototypen kompatibel med skärmläsare.

Nyckelord: sonifiering, mappningar, auditiva egenskaper, användbarhet, synnedsättning, skärmläsare

Acknowledgements

First I would like to thank my supervisor Kirsten Rasmus-Gröhn at the Department of Design Sciences, LTH, for guiding me throughout the whole thesis process and providing valuable feedback. Secondly I would like to acknowledge the test participants for their time and their important insights that made this thesis possible.

Contents

1	Introduction	4
1.1	Background	4
1.1.1	Sonification and Sound Data	4
1.1.2	Screen-Reader in Relation to Sonification	5
1.1.3	Usability and User-Centered Design	7
1.2	Related Work	8
1.2.1	Existing sonification tools	8
1.2.2	Basis for Prototype	8
1.3	Purpose and Research Questions	8
1.4	Scope and Delimitations	9
2	Methodology	10
2.1	Tool Study	10
2.1.1	Desmos	10
2.1.2	Sonification Sandbox	11
2.2	Initial Usability Test	12
2.2.1	Setting Up the First Usability Test: Steps and Procedures	14
2.2.2	Lessons Learned From the Pilot Test	15
2.3	Prototype	15
2.3.1	Early Development	16
2.3.2	JupyterLab Prototype	16
2.4	Second Usability Test	18
2.4.1	Setting Up the Second Usability Test: Steps and Procedures	19
2.4.2	Test Improvements Discovered by the Pilot Test	19
3	Results	21
3.1	Initial Usability Study	21
3.2	Second Usability Study	22
3.3	Final Prototype	24
3.3.1	Visual aspects	24

3.3.2	Functionalities	25
4	Discussion	28
4.1	Usability tests and Tool Study	28
4.2	Test Results	29
4.2.1	First Test	29
4.2.2	Second Test	30
4.2.3	Research Questions	32
4.3	Prototype	34
4.4	Future Work	35
5	Conclusion	36
	References	37
	Appendix A Test-Interview Answers	40
A.1	Initial Usability Test	40
A.2	Second Usability Test	43
	Appendix B Prototype	46

Chapter 1

Introduction

In this chapter the relevant background to the thesis is presented, related work is explored, purpose and research questions are introduced, and the scope and delimitations of the thesis are discussed.

1.1 Background

Sonification is the idea of turning different types of data into sound, for example a plotted graph of data points where each point produces a sound based on its value. The motive for this exam work is to explore the world of sonification, to look at parameters that can distinguish different data points more efficiently and to design a user-centered prototype where the users are people with visual impairment.

1.1.1 Sonification and Sound Data

When this report discusses Sonification it is the idea of translating different types of data into sound [1], where the visual example could be a typical Cartesian graph with plotted values where the plotted values produce a designated sound based on selected settings. By translating information into sound, changes in frequency, pitch, amplitude, and location in the stereo field reveal changing variables to the listener [2]. When talking about parameters such as pitch, note duration, and note velocity, these will be called auditory characteristics throughout this report. When data sonification is in a musical context it can facilitate or augment the learning process; studies reviewed in [3] have found that passive and active music listening improve the performance of several cognitive tasks such as reading comprehension and memory. A clear use-case for data sonification beyond the cognitive positives is its interpretability for those with visual impairments who may not be able to reliably decipher traditional data visualizations [2].

There are four different groups of data appropriate for sonification, these are: sound recording data, general acoustical data, physical data, and abstract data. Sound recordings is normally sampled digitally as a series of numbers that produce a continuous sound signal. The point of sonifying sound data is that it can be amplified, and/or time-compressed or -stretched to enable sounds otherwise inaudible by the human ear to be heard such as bat ultrasonic calls. Sonification in this context could therefore be used as an acoustic micro- or telescope [1].

General acoustic data include all kinds of measurements in elastomechanics such as vibrational data of mechanical waves that become accessible by listening to their sonification. A recognizable example of this is using a stethoscope to listen to internal body sounds. The character of mechanical waves is usually preserved when it is reduced to a one-dimensional audio signal and when changing playback speed of the waves there is only a minor deterioration of its resulting sound [1].

Physical data are the measurements of other physical processes outside the mechanical domain that can be sonified but the data usually lacks acoustic familiarity with daily hearing. An example would be Electroencephalography (EEG) data of several electrodes around a head which cannot be compared to a similar arrangement of microphones within a room so when interpreting such data one must be mindful of these limitations [1].

Abstract data has an even lower acoustic familiarity than the physical data due to its abstract nature, it is however quite easy to sonify when the abstract data is arranged as a time series. An example of abstract data could be stock market data, fax-machine noises or a computer-modem. Not all wave-like shapes in abstract data conform to the wave equation, so interpreting those sonified signals is usually harder for the user to get accustomed to [1].

Regardless of which type of data that will be sonified, the data that will be used in the thesis is digitalized. In order to sonify the digitized data it will be mapped towards Musical Instrument Digital Interface (MIDI) output which determines what the data will sound like. One of the motives of this thesis is to determine appropriate mappings between the data and the MIDI output and to tailor the sound towards this thesis' end user: people having some kind of visual impairment. While adapting the sound for people with vision impairment is the main focus it would be seen as a great bonus if the sound would be a valuable complement to visual data.

1.1.2 Screen-Reader in Relation to Sonification

Screen-readers are software programs that help people with visual impairments to use technology such as computers and smartphones. The screen-readers read text that appears on screen through a speech synthesizer and can by different commands perform lots of different tasks, examples are finding a string of text on the screen, or announcing the location of the computer's cursor or focused item. It can also perform more advanced functions such as locating text displayed in a certain color, reading pre-designated parts of the screen on demand, reading highlighted text, and identifying the active choice in a menu. All in all, a screen-reader acts as an interface between an operating system, applications, and the user [4].

Web accessibility guidelines, such as WCAG [5], recommend alternative representations such as image descriptions in the shape of alt text, and tabular data, to make web content accessible to screen reader users, which is also the most common way of doing it. Descriptions, and other textual representations, have a drawback in that they don't allow for direct access

to the underlying data. Rather than assisting readers in forming their own interpretations, they simply reflect the author's viewpoint. This limitation can hinder a reader's ability to fully engage with the information presented. With data that is automatically updated, descriptions can become antiquated which creates an imbalance between the description and the data that is visually presented. Maintaining high-quality descriptions can be a challenge, especially when attempting to convey complex spatial information through graphics. This presents a cognitive burden on the user, making it difficult for them to fully understand the message being conveyed. These issues with descriptions is why sonification is an alternative that provides a more direct access to the data [6]. Data narratives are written summaries of data sets that draws conclusions, makes comparisons, and in detail explains the meaning of the data. Well-designed data narratives helps users discover important trends, draw comparisons and see the relevance and importance of the data regarding the users' topic(s) [7]. Data narratives have mostly been applied to visual graphics but these narrative techniques can be extended and applied to audio graphics as well [6]. A combination of sonification and data narratives could therefore prove to be efficient and shows that there is a place for both techniques in aiding visually impaired users. In the study performed by Siu et al. [ibid] they found that interleaving descriptions with sonification, which is called audio data narrative representation, helped their screen-reader users to gain a more complete understanding of the data and it was more efficient than presenting description followed by the sonification a.k.a. control representation. To simplify the difference between audio data narrative- and control representation figure 1.1 was drawn based on figure found in [ibid]. The audio data narratives are extra useful for complex data sets with more than two trend reversals but the benefit lessens with more simple data sets [ibid].

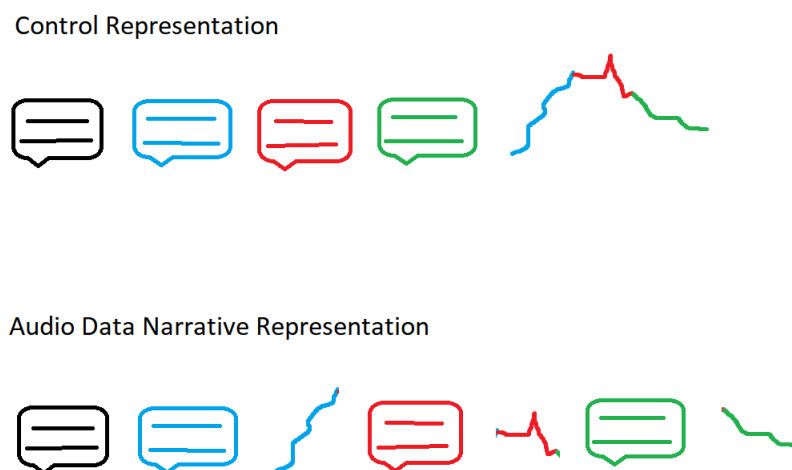


Figure 1.1: Control- vs. audio data narrative representation

Redrawn from Alexa Siu, Gene S-H Kim, Sile O'Modhrain, and Sean Follmer. Supporting accessible data visualization through audio data narratives. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

1.1.3 Usability and User-Centered Design

In short what makes something usable is the *absence of frustration* in using it which is something that usability is defined as. Usability can also be divided into six different attributes: useful, efficient, effective, satisfying, learnable, and accessible [8].

Usefulness is the degree to which a product enables a user to achieve its goals and the willingness of the user to use the product. Without usefulness it does not matter if the product is easy to use, easy to learn and satisfying to use because it does not achieve the goals of the user and therefore will not be used [8].

Efficiency pertains to the quickness of which the user's goal can be accomplished accurately and completely and it is often measured in terms of time. For example an efficiency benchmark in a photograph application could be "90% of users will be able to find the zoom function within a minute if asked to" [8].

Effectiveness is the extent to which the product works the way the user expects it to and how easy it is for the user to do what they intend. Effectiveness can be quantitatively measured through error rate such as "80% of users will be able to zoom correctly according to instructions on their first attempt" [8].

Learnability is part of the effectiveness in that a user should have some level of competence after a predetermined amount of time of using the product. It can also refer to sporadic users ability to relearn the system after a time of inactivity [8].

Satisfaction involves the user's perceptions, feelings, and opinions of the product after using it. If the other attributes of usability have been received positively the satisfaction tends to be higher and if the other attributes have not been considered during implementation the satisfaction tends to be lower [8].

Accessibility and usability are siblings because accessibility is about having access to the products that are needed for a specific goal but it can also be in terms of how accessible the product is to people with disabilities [8]. Especially in this thesis, the second interpretation is what is of importance due to the intended end user being people with visual impairment.

Both usability and accessibility are part of the larger discipline of user-centered design (UCD) which represents the techniques, processes, methods, and procedures that are used for designing products and systems that are usable; but maybe even more importantly it places the user at the center of the process. When a team is about to build a product the technology and features of the product are the first building blocks, but the team should also imagine who the end user will be in order to fulfill the usability attributes. UCD seeks to support how target users will use a product instead of forcing those users to change how they would use a product. According to The International Organization for Standardization (ISO) in standard 13407 UCD is "characterized by: the active involvement of users and a clear understanding of user and task requirements; an appropriate allocation of function between users and technology; the iteration of design solutions; multidisciplinary design.". In short, the three basic principle of UCD are:

- Early focus on users and their tasks
- Evaluation and measurement of product usage
- Iterated design [8]

1.2 Related Work

From previous chapters we learned what sonification is and what kind of data can be used in sonification which is discussed by Hermann et al [1]. When it comes to cognition and learning, the subject of sonification appears to be well studied [2, 3]. The usage of sonification for the visually impaired is researched but the focus of that research is usually assistance with daily life and navigation [9, 10] and sonifying data, which is what this report is exploring, is only investigated at a small sample-size such as in [11, 12, 13]. Something that is not thoroughly researched within the sonification field is the usability aspect, especially in combination with visual impairment, however it is investigated in an article by Sharif et al [14], where they let screen-reader users evaluate the usability of different sonification-prototypes based on pleasantness, clarity, confidence and overall score. Investigating different types of auditory characteristics such as pitch, volume, timbre, and panning and how people with visual impairment perceives these channels is something that is done in [15] and could act as a comparison to the findings of the usability tests that are performed during this thesis.

1.2.1 Existing sonification tools

While there is lots of research regarding sonification, what kind of tools are out there that can take data and sonify it? Early on in the thesis project two different tools were identified and a decision to investigate these two tools was made. Investigating the two tools and comparing them aims to bring useful data for the eventual making of the sonification prototype. Both what these two tools can accomplish in regards to sonification and what they might be missing will be taken into consideration while performing the analysis.

1.2.2 Basis for Prototype

Along with the knowledge of how existing prototypes work, there are more baselines that will be followed when it comes to the prototype. In the video Sonification with Python Dr. Matt Russo guides you through the process of turning data from a CSV file into a sonified graph through a number of steps inside JupyterLab [16]. The video's way of taking data and producing sonified graphs is the basis for the prototype and additional functionalities will be figured out through usability tests and prototyping.

1.3 Purpose and Research Questions

The purpose of this thesis is to figure out sonified data's optimal auditory characteristics of pitch, note duration, and note velocity and to apply these to a user-centered prototype where the end users are people with visual impairment. Furthermore, the thesis will investigate how to combine the use of sonification and a screen-reader to evaluate data in an efficient and user-centered way.

This paper aims to answer the following research questions:

1. How should auditory characteristics be configured to enable smooth interpretation of sonified data?

2. How do you design a software prototype for someone with visual impairment?
3. In what ways can the end user interact with the sonified data through the prototype?

1.4 Scope and Delimitations

The prototype is developed with the goal of exploring the research questions and not being a fully functioning product. The development time of the prototype was also quite limited which means that simpler solutions are chosen and design decisions are made accordingly. The prototype was developed through JupyterLab so the technical solutions and testing opportunities of the prototype are therefore limited to the constraints of JupyterLab. Only one test participant which fits the end user criteria was included in the testing, the reason being that tech-savvy visually impaired people that are also interested in helping this thesis are hard to find. Additional test information could instead be gathered through so called imitated end users.

Chapter 2

Methodology

The methodology chapter outlines the different methodologies that are approached during the thesis work. There is an overall focus on usability and UCD enabling an iterative workflow when prototyping. A tool study and usability tests are performed and insights discovered are then used in the iterative workflow.

2.1 Tool Study

The two tools that were investigated during the tool study are Desmos and Sonification Sandbox. The reason for investigating the two tools were to find positives and negatives that could act as guidance for the prototype and to select one of the tools to use in the first usability test.

2.1.1 Desmos

Desmos is an advanced graphing calculator that is implemented both as a web application and as a mobile application. The web application will be the one analysed. While the main use of the application is to graph equations and perform a long list of mathematical functions it can also sonify the graphs. As long as there is a corresponding graph to be seen, the data can be sonified; examples are expressions, functions, and distributions. An example of such a graph can be seen in figure 2.1. Individual points in a graph can be listened to, but the points are chosen automatically by the application and not manually and the minimum euclidean distance [17] between the different points is constant. The sound being played is mapped in a way so that the higher up on the y-axis it is, the higher the pitch; a pan (stereo sound) from left to right is played depending on if the value is at the left or the right side of the currently viewed x-axis. There is also a white noise added to the sound whenever the point is below 0 in the y-axis. If several graphs are present in the diagram only one can be listened to at the same time, but if two graphs intersect a "popping" sound will be played as an indication.

All the sounds being sonified are relative to the screen which means that points outside the current view will not be sonified and the speed of the sound depends on the relative size of the graph compared to the overall view. Because of the sound being relative to the screen by zooming in or out the user can make the sound longer or shorter respectively. Another functionality of zooming is that the user can select certain segments of a graph that should be sonified. Settings related to sonification are limited to changing the speed and the volume of the sound; the volume can change between 0% and 100% with a 10% increment and the speed can change between $\frac{1}{4}x$, $\frac{1}{2}x$, $1x$, $2x$, and $4x$ speed. Desmos also has a screen-reader function that should be able to describe a point, a curve or the axes if a screen-reader is turned on in the unit it is used on, but the function did not work while using Windows' "Narrator" in the web application.

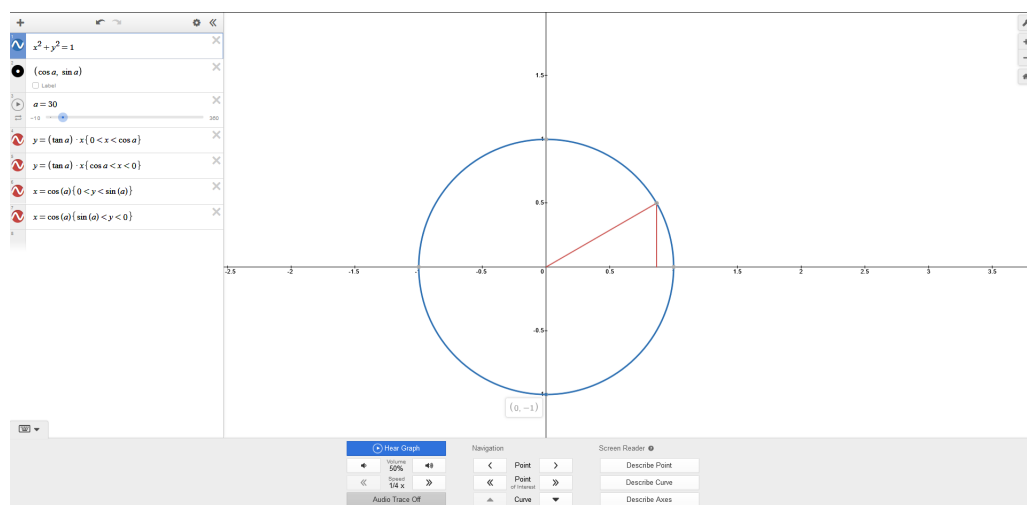


Figure 2.1: A unit circle in Desmos with sonification options at the bottom

When it comes to Desmos' limitations the biggest one is that a user cannot import data into the application which means that graphs will have to be created manually; the application does have some finished example graphs to use, but they are limited. Another big limitation is that mappings cannot be changed i.e auditory characteristics such as pitches, instruments, time played, and panning. While individual points can be played, the user cannot specify exact points of which they want to sonify.

2.1.2 Sonification Sandbox

Sonification Sandbox is a tool built to sonify data, so unlike Desmos it is the primary function of the application. The application is downloaded as an executable file and the latest version (6.2beta) is the one being analysed. Data in the application is inserted into columns in a spreadsheet and the application also allows the user to import data from CSV files which in turn allows a user to edit the data separately in Microsoft Excel or equivalent program. The import function does not work in the current version, however the data can still be copied and pasted from the separate program. If the user does not want to use a separate program some quick editing of the data can be performed in the application, where the data can be selected, cut, copied, pasted, and cleared. The columns can also be filled using simple

equations such as $A + B$ where A and B are column A and B in the spreadsheet. The visual diagram has column values on the y-axis and column A decides the time-values on the x-axis. The user can listen to several columns simultaneously but there is no indication whether they intersect or not. In contrast to Desmos there are more mapping options to explore in Sonification Sandbox, there are both global- and column specific settings. Global settings determines how long the graphs will play time wise and there are no limitations, it could be one second, but it could also be a year for example. Specific column settings allows the user to select pitch, volume, and panning; the manner of which they can be changed can be seen in figure 2.2. The user can apply contextual settings to data context (y-axis) and to time context (x-axis). For the data context the user can add a constant or repeating tone that will play at either minimum, maximum, or mean value. Additionally the instrument, pan, pitch, and volume can be tweaked for the data context. In terms of time context the user can choose the beat, frequency of the beat, instrument, pan, and pitch; figure 2.3 will clarify what that could look like. Just like Desmos, Sonification Sandbox has a screen-reader function that can explain columns or the whole diagram, but just like Desmos it did not work in version 6.2beta while using Windows' Narrator.

While Sonification Sandbox offers some additional mappings and settings compared to Desmos it is still lacking in some areas. Unlike Desmos the application cannot select individual points and/or graph segments to be sonified which is one of the biggest limitations. Another big limitation is if two or more columns are to be played simultaneously they need to be close to each other in value. If one column has values in the 100s and another column has decimal values both of them will not be displayed in the diagram, and therefore cannot be sonified. Some smaller let-downs of the application is that the fill column function is quite limited, which can be circumvented by editing data in a separate program and then import it, and that the application will freeze from time to time which forces the user to be meticulous in its saving. There seems to be a learning curve to use the program and setting up a smooth graph using the values personally chosen could not be achieved. Based on the difficulty of using Sonification Sandbox, Desmos is the tool of choice for the initial usability study that can be read about in 2.2.

2.2 Initial Usability Test

In order for the prototype to be developed and for it to be tailored towards the end user it was decided that an initial test should be performed. The initial test would be using an existing tool, which was Desmos in this case in order to gain a base understanding for how a tool would be used and what areas that may need improvement or are missing entirely. By observing a test participant, pain points or general design improvements were discovered and by asking follow up questions the feedback needed for the prototype was collected.

The idea was to first test the test by having a willing test participant simulate blindness by using a blindfold. The test was planned in the same way as the "real" test, but with consideration of the inexperience using a screen-reader, which meant that a bit more intervention during the test was allowed and expected. The reason for performing the test on an imitated end user was to rehearse how the test would be performed, but the imitated test, or "Pilot Test", would also provide additional feedback that could strengthen the conclusions made from performing the actual end user test. Both the pilot test and the "real" test were per-

Global Settings
Column B settings

Column B Settings ()

Enable Rendering for this data series

Sound scales

Use Global Settings

Pitch

Enable Pitch Mapping

Timbre: Piano

Default: Note #84 (C)

Min: Note #56 (Ab)

Max: Note #113 (F)

Polarity: Positive Negative

Pitch: Round Exact

Interpolation: On Off

Pan - stereo position of the sound, left-to-right

Sweep Default: 1° left

Enable Pan Mapping

Min: 90° left

Max: 89° right

Polarity: Positive Negative

Volume - the loudness of this line

Enable Volume Mapping

Default: 100%

Min: 0%

Max: 100%

Polarity: Positive Negative

Visual graph settings

Enable Graph Mapping

Line Color: Automatic Select Color

Stroke Type: Line

Time Context
Value Context

Value Context

For Line: 1 - (no name) play the min this often: constant

Instrument: Piano

Pan: -27

Pitch: 60

Volume: 100%

Time Context
Value Context

Time Context

Beat Type: BEAT

Beat Frequency: 1

Instrument: Piano

Pan: 0

Pitch: 60

Figure 2.3: Value context- and time context settings

formed in Swedish but all of the answers and questions were translated into English for the readability of this report. Both tests were performed in a "home" environment i.e both test participants were at their respective homes when participating in the test.

2.2.1 Setting Up the First Usability Test: Steps and Procedures

In order to set up the initial usability test a few encompassing questions were asked:

- What data should be used?
- What information should the test produce?
- How should the test be performed?
- How is the test useful for prototyping?

With these questions in mind it was decided that the test should incorporate three different tasks: the test participant will have to listen to two sonified mathematical graphs, $y = kx$ and $5x^3$, as well as listening to a sonified histogram containing bars at different heights including 0, see Figure 2.4.

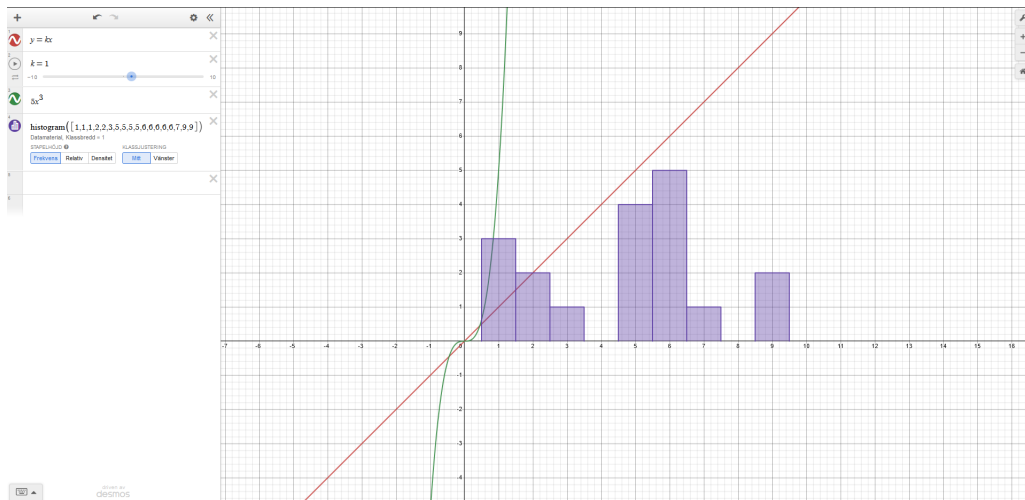


Figure 2.4: Figure showing what the different tasks look like. (Red and Green lines are zoomed in during the test)

The general procedure of the test was:

1. Let the user listen to the tasks → ask questions about how well the user understand what they are listening too and what would make the tasks easier to understand via sonification.
2. Let the user listen to the second task at different speeds → ask question about what they prefer.

3. Let the user listen to individual points of the second task → ask if they can identify whether the point is below the x-axis or not.
4. Let the user listen to a screen-reader explaining the graph → ask several different questions regarding the use of screen-reader both on its own and in tandem with the sonification.

While the usability test is pre-planned and will be followed through as closely to plan as possible, improvisation will be allowed if needed to get a conversation going or to make things clearer for the test participant, quite similar to a semi-structured interview. When it comes to the prototype it is mainly questions from step 1 and from step 4 of the test that will provide the substantial feedback needed to make decisions on the prototype's functionality but the others also bring important information. The focus when looking at the screen-reader is what kind of information it is giving the user rather than how the user navigates using the screen-reader. Additional details such as verbatim questions and notes from answers can be found in A.1.

2.2.2 Lessons Learned From the Pilot Test

Apart from gathering valuable extra answers to the test questions a few different improvements to the test were discovered and were applied to the "real" usability test. The following test changes were identified:

- After listening to a task I would ask three different questions, the first two were very alike so I decided to remove one of them because no new information was gathered from asking them both.
- I should remind test participant to focus and listen as well as letting them know when I am starting the sound sequence.
- When explaining what the test participant is listening to (explained before asking what would make task easier to understand) language should be adapted to the test participant's math knowledge.
- It would be easier if a speaker (with panning) is used in the test because I can hear what they are hearing and identify when things are not going according to plan.
- I changed the phrasing of the last question regarding the screen-reader- and sonification relationship.

2.3 Prototype

The overarching goals of the prototype was to develop a tool that can sonify data and to make the tool interactable and usable for people with visual impairment. For prototype navigation, the usage of a screen-reader and keyboard shortcuts was assumed.

2.3.1 Early Development

As mentioned in 1.2.2 the starting point of the prototype came from a YouTube guide that shows how to transform data from a CSV file into a sonified MIDI file in JupyterLab. The initial idea was to make a React.js web-page that displays data in a graph with accompanying buttons for playing the sound as well as some other interactions with the data. The reason to use a web-page was because users are familiar with browsers and that it could easily be kept minimalistic. The functionalities of the page would come from the Python code in JupyterLab. Eventually a realization that the web-page idea would be very complex when it came to linking Python code with React.js was made. It was therefore decided that the prototype should be made entirely within JupyterLab which has an easy-to-use interface when working with visuals and Python. Before changing to JupyterLab a simple web-site was built which you can see in figure 2.5.

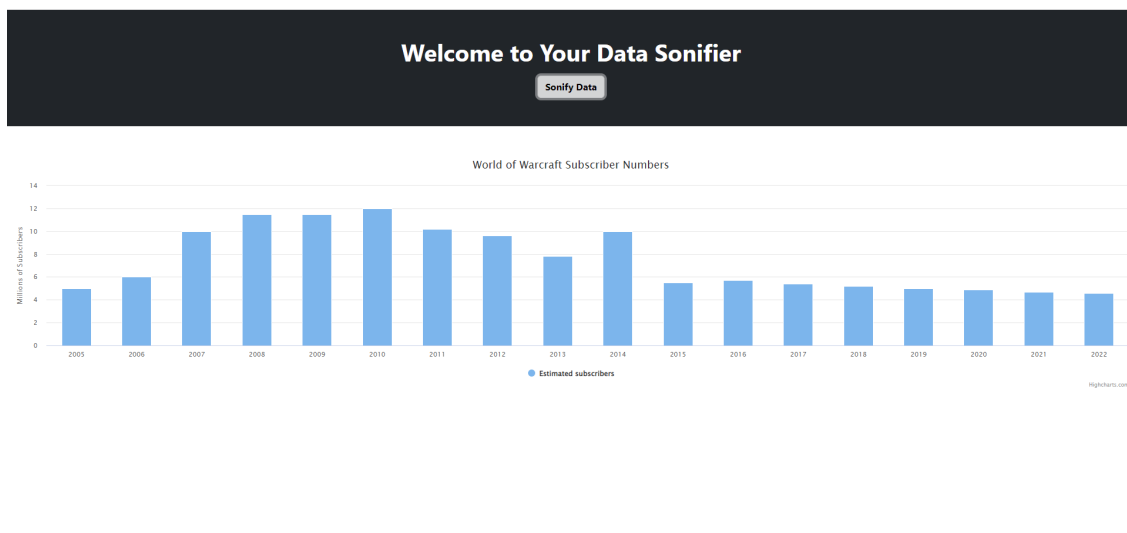


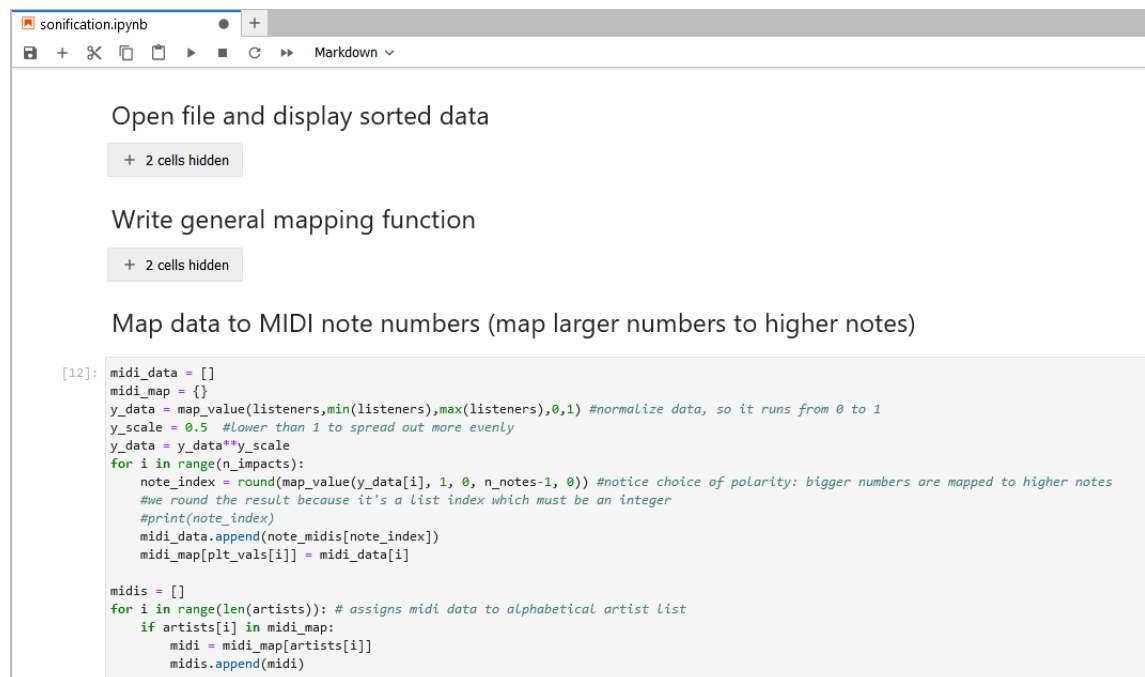
Figure 2.5: Simple React-page with example data and a button

2.3.2 JupyterLab Prototype

In the YouTube guide, Russo used data consisting of lunar craters. These lunar craters had names, longitude, latitude, diameter, and age values and Russo used their ages and diameters to plot the data and then through a number of steps mapped MIDI note numbers to these data points and got a resulting sonification. In order to differentiate from Russo's program another data set was chosen: "Top 100 artists on Spotify based on monthly listeners". The data was retrieved from [kworkb¹](https://kworkb.net/spotify/listeners.html) and was manually transferred to a CSV file with two columns: artist name and monthly listeners where the monthly listeners were rounded and then entered as millions of listeners. Having a data set where one of the columns consisted of names meant that there was only one column of numerical values to use whereas Russo used two columns of numerical values in the making of his program which naturally changed the process of CSV → MIDI file.

¹<https://kworkb.net/spotify/listeners.html>

First of all the data was sorted alphabetically so that the resulting sonification would have some variation and not go from lowest to highest value or vice versa. During the mapping of MIDI note numbers to data the larger numbers were mapped to lower pitched notes and the note velocity was assigned so that larger number equaled a greater note velocity; note velocity is the combination of volume and intensity and can be compared to hitting a piano key with larger velocity which produces a louder and more intense sound. Pitch was set by MIDI note numbers and ranged from 'C1' to 'A6'. Note velocity was set by MIDI velocity which are integers ranging from 0 to 127, the range chosen for the prototype were 35 to 127. How the pitch was assigned can also be seen in figure 2.6 The complexity of these mappings with the Spotify data set were that the data needed to be sorted by monthly listeners during the mapping and afterwards regain the alphabetical order before creating the MIDI file. It was solved by saving the mappings in a python-dictionary and then locating the values by their keys (artist names). MIDI file was then created with pitch and velocity values from the mappings. The intention of the MIDI file was to play it in JupyterLab through the pygame² package but it was discovered that the MIDI format was not working and the file had to be converted into a mp3 file in order for it to work.



The screenshot shows a JupyterLab notebook interface with the following sections:

- Open file and display sorted data**: A button labeled "+ 2 cells hidden".
- Write general mapping function**: A button labeled "+ 2 cells hidden".
- Map data to MIDI note numbers (map larger numbers to higher notes)**: A code cell containing the following Python code:


```
[12]: midi_data = []
      midi_map = {}
      y_data = map_value(listeners, min(listeners), max(listeners), 0, 1) #normalize data, so it runs from 0 to 1
      y_scale = 0.5 #Lower than 1 to spread out more evenly
      y_data = y_data**y_scale
      for i in range(n_impacts):
          note_index = round(map_value(y_data[i], 1, 0, n_notes-1, 0)) #notice choice of polarity: bigger numbers are mapped to higher notes
          #we round the result because it's a list index which must be an integer
          #print(note_index)
          midi_data.append(note_midis[note_index])
          midi_map[plt_vals[i]] = midi_data[i]

      midis = []
      for i in range(len(artists)): # assigns midi data to alphabetical artist list
          if artists[i] in midi_map:
              midi = midi_map[artists[i]]
              midis.append(midi)
```

Figure 2.6: Code example showing how the midi numbers were assigned, but also how the segments and development within JupyterLab can look like

After being able to play the file within JupyterLab the next step was to create some kind of user interface (UI) where a user would be able to interact with the sonification and gain information. The prototype went through a lot of iterations and originally the prototype displayed the plot of the 100 artists and their monthly listeners and the UI consisted of two sliders and two buttons, the sliders controlled the starting position of the playback and the volume while the buttons started and stopped the sonification. The starting position refers

²<https://www.pygame.org/docs/>

to which data point the sonification starts from, the slider goes from 1 to 100 since there are 100 data points. Both the position slider and the volume slider can be used while the sonification is playing and dynamically change the playback, so as an example the user could drag the slider to "5" and the playback will rewind and then start at the fifth point with little-to no delay even though it might be playing at point eight already.

For the next iteration the introduction of screen-reader compatibility was supposed to be added; some of the elements such as the buttons and sliders were read by the windows screen-reader that was used while prototyping but a diagram description was missing. The idea was to force a screen-reader to recognize the plot as an object that could be read which proved hard to do, and in the end it was decided that a "hard coded" diagram description would suffice. Using the python package "accessible_output2" a standard female robot voice reads whatever text it is told to read and through that a description of the diagram was added.

After some feedback from my supervisor the changes that were suggested were:

- Make the two sliders as wide as possible to simulate it being a physical slider a bit better
- Make sonification play 10 notes at a time based on the positional slider for analysing a part of the data without needing to pause
- Try to make midi playable within JupyterLab which would enable more dynamic functionalities to be added because the file converting can be skipped

It was also discovered that higher values were assigned to lower pitches although the intention from the beginning was to assign it the other way around i.e higher values to higher pitches, which was fixed. Increasing the size of the sliders also proved to be an easy task as the sliders had a setting for size and it made sense to change. Making the sonification play 10 notes at a time could be accomplished in several different ways but in all of these solutions that were tried it blocked other functionalities from working; as an example using a `time.sleep()` that stops after 10 seconds let 10 notes be played before it stopped (each note was 1 second long), but then the user could not change the positional slider dynamically or stop the sonification. Several solutions to that issue were tried, such as using threading but none of them seemed to work as intended and in the end this suggested change was postponed to a future update. The third suggested change was also postponed after lots of trial and error, trying different python packages than `pygame`, or using `pygame`'s `midi` module but the issue seemed to lie within JupyterLab compatibility and not `pygame` itself. During this stage of development some quality of life (QoL) changes were also added, mainly a toggle box that would toggle the diagram description from being read or not but also lots of code cleanup. At this stage the prototype was the version used for the second usability test and it is also where the development stops in this thesis work.

2.4 Second Usability Test

The first test was done to help in developing a prototype while the second usability test was instead done to improve on the prototype that was produced based on that first test's insights. Besides getting feedback on the produced prototype the second usability test provides helpful information for this thesis' research questions.

Just like the first usability test, the second one was also performed on both an imitated user using a blindfold and a real user for the same reasons as the first test. Additionally the second round of testing was performed in Swedish and afterwards translated to English for the sake of the report.

2.4.1 Setting Up the Second Usability Test: Steps and Procedures

Very much like the first usability test some questions were asked to help in planning the test, but instead of four different questions only three were needed for this test. The question "What data should be used?" was already answered because the prototype consists of one data set. The rest of the questions remained the same, however "How is the test useful for prototyping?" changed to "How does the test help in improving the prototype and answer the research questions?"

Taking these questions into account, the following structure of the test was determined:

1. Let the user listen to the whole graph with graph explanation enabled → ask questions about the user's apprehension of the sonification and the aptness of the graph explanation.
2. Let the user listen to the last 10 points of the graph (graph explanation optional) → ask the user to convert what was heard into a graph and ask them to compare the lows and highs of the 10 points with the rest of the graph.
3. Ask if auditory characteristics were appropriately assigned → let the user listen as they see fit in order to answer.
4. Let the user play around with the prototype until they feel that they have explored everything → ask questions about current- and potential future functionalities.

Just like the first usability test, this test had a plan in place with the option to improvise as seen fit in order for the test participant to understand their task and/or to retrieve relevant information. The first three steps of the test are related to this thesis' first research question and the fourth step as well as the whole test explores the other two research questions, see 1.3. A more detailed overview of the questions and answers can be found in A.2

2.4.2 Test Improvements Discovered by the Pilot Test

After performing the imitated test there were lots of improvements discovered which was later on implemented before performing the test with the actual end user. Feedback points and improvements are presented below:

- After listening to the whole sonification the consensus between me and the imitated user was that listening to 20 points instead of the whole 100 points would be more efficient while not really losing any insights. Another feedback point was that the "graph explanation" toggle should really be named "diagram explanation" or similar because the data points are displayed in a diagram and not a graph. In the end it was named "Diagram Description".

- Decided to rephrase one of the questions in step two to "Describe the pattern of the last 10 notes" because it is easier to convert that answer into text. Also rephrased the other question in step two a bit, it was said that the user should compare with the rest of the sonification points but since 20 points felt like enough in step one it was decided that the last 10 points should be compared to the first 20 instead.

Chapter 3

Results

3.1 Initial Usability Study

Let us start by establishing two acronyms Imitated User (IU) and Actual User (AU) who were the two test participants that the first usability test was performed with. The first usability was performed two times, one per test participant, and through these two tests some insightful feedback to bring along to the prototyping was gathered. After both users had listened to the three different tasks one by one and then explained what they thought they heard for each one, the results are that they could interpret what they were listening to pretty well. They both understood the general direction and value change of the data but the resulting explanation could only serve as an approximation of the task they were listening to. The third task that consisted of a bar diagram was the easiest one for both the users to interpret accurately, however the actual numbers of the different bars could not be conveyed, only the relationship between them in terms of height which is also something that AU answered in one of the interview questions (see A.1).

The second interview question asked for changes to the sound of the different tasks. IU suggested that the pitch should be lower overall for the first two tasks, that the static sound below the x-axis makes things harder to interpret (without having the context of why it was there), and the third task should have had a higher pitch overall or a more relatable sound such as a piano. IU also believed that a non-continuous sound would be easier to interpret than the continuous ones of task- 1 and 2 i.e small pauses between values in the curve. AU agreed with IU regarding the static sound; it disrupted the interpretation of the graph. AU thought there should be longer pauses between the values being played in the bar-diagram since it is not a continuous graph. After learning what the static sound was, both test participants agreed that it was a good feature which adds information to the sonification.

When it came to the speed of the sonification the consensus between the test participants was that $\frac{1}{2}x$ speed was the preferable speed for the task it was tested on (task 2). IU said that the "flat" part of the graph was easier to pinpoint at that speed compared to the others. AU

said that $\frac{1}{2}x$ speed should be the default speed due to being able to see the "full picture" of the graph while it is also easily interpreted when it comes to acceleration. AU also saw uses for the other speeds though, $\frac{1}{4}x$ speed would be good when analysing very complex graphs and the faster speeds of $2x$ and $4x$ would be good when analysing longer graphs.

The second part of the usability test investigated the usage of a screen-reader and what kind of information it provides. Both IU and AU agreed that there is enough information given by the screen-reader but AU mentioned that the interface and navigation of Desmos was difficult. The test participants were asked if they received information from the screen-reader that they did not get by listening to the sonification and they both said that the boundary values of the two axes were only given by the screen-reader. AU addressed that the only information given "value-wise" by the sonification was whether the value was below 0 or not in the y-axis. IU also mentioned that the exact formula ($5x^3$ in this case) was given by the screen-reader, which was something that AU heard too. When asked how the screen-reader should be used in conjunction with the sonification, both users said that the screen-reader should automatically read the information before the sonification sound is playing. IU argued that having it read during the sonification would be distracting but AU said that it might work if the sonification was playing at slower speeds and having it as an option might bring value to the prototype. IU also mentioned that having the screen-reader read its information after the sonification would make you want to listen to the sonification again to confirm the information you received by the screen-reader. Having a keyboard shortcut that plays the screen-reader could be an alternative but having it automatically play before the sonification makes them stick together better according to IU while AU said that having a keyboard shortcut as an option would be nice. How to balance things between the sonification and the screen-reader was also something that was asked to the test participants and they both thought that the current dynamic between them was healthy. AU did mention that he thought the sonification could bring more information such as having different tones for different colors in a bar diagram though.

AU was very helpful in providing general feedback when it comes to designing things for people with visual impairment, some pointers of what should be considered during prototyping were:

- Keyboard shortcuts are good
- Tabbing is usually how you navigate a page or an app which means they should have a minimalist design
- Screen-readers have two modes when navigating a HTML page, editing- and reading mode
- Enter usually means "mouse-click", alt usually is the menu button etc. Look up standards of screen-reader navigation
- Putting titles to different sections of the application helps with navigation.

3.2 Second Usability Study

This section will be written continuing with the two acronyms IU and AU representing the two test participants who remain the same from the last usability test. Early on for both

test participants it became quite clear that navigating within JupyterLab without visual aids was inefficient as the test participants had to "tab" through several elements that were not important to the prototype in order to reach the prototype functionalities. After listening to the sonified data both IU and AU said they could understand the difference in values and IU could tell the general range of the values while AU suggested that there should be audible reference values to help the user understand how high or low the pitch can go. A diagram description was also read out aloud by a typical robot voice before the sonification was played. Both test participants thought the information given by the diagram description was useful but AU thought that the reference values could be included in the description and IU thought it was a bit fast and some numbers that was described could have been rounded for simplicity.

In the second part of the test IU and AU were told to listen to the last 10 values of the sonified data and then describe the pattern they heard and the pattern they were trying to hear looked like figure 3.1. The IU described it as "Starts at low pitch then moves in a zigzag pattern and stops at an even lower pitch than the start." and the AU said "First one was low, one value was pretty high, among the last ones there were some that were the same and one that was very low. The highest value was close to the end.". From these statements one can tell that they both had a good understanding of what they heard even though they missed a few things. The test participants were then asked to compare these 10 values to the values they listened to in the beginning of the test and both users recognized that the highest value among the last 10 were higher than the values they heard before, IU could also hear that the lowest value among the last 10 were lower than most other values while AU believed the lowest value was harder to compare with the first 20.

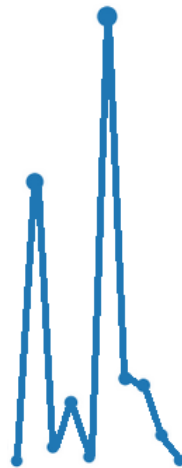


Figure 3.1: The ten last values of the sonified data

Next on the test agenda was asking questions about the configuration of auditory characteristics, namely pitch, note duration, and note velocity. Both users said that the way the pitch was assigned was very logical and AU added that if the data was in a graph format it should be continuous like in the previous usability test. IU had the opinion that both note duration and note velocity were appropriately assigned to the values and had no further comments about it. AU thought that the note duration was a good standard speed and mentioned that it could be even faster with more experience listening, AU also added that it might have been nice to have it even slower for a first time listener. As a final point regarding the note

duration AU said that there should be an option to change the speed of the sonification. When it came to the note velocity, AU acknowledged that it provided more realism to the piano sound, which sounded nice. However, AU did not believe that it added much regarding the apprehension of the sound. AU thought that pitch was more important in that regard.

Lastly the test participants received questions about the prototype functionalities. When asked about missing functionality IU could not think of anything in particular that was missing from the prototype. AU wished that the prototype would be more screen-reader compatible since it did not read everything automatically which AU noted might be a windows screen-reader limitation. AU also mentioned that if graph data was used it should have a 0-value divider and be continuous just like in Desmos. A fun to have functionality would be switching between different instruments, AU said that the piano sounded pleasant and high- and low pitches were easy to interpret, but it might not be preferable for everyone. AU mentioned that a speed changer would be nice to have when asked about the note duration earlier in the test and AU said it would be nice to have because after a while you want the speed to be faster and sometimes when something is hard to understand you want to slow it down. AU also wanted to see key-binds for the sliders and the buttons. When asked if the current functionalities were unnecessary or needed to change IU said that the functionalities felt necessary and did not need to change. AU said that the interface was simple and hard to mess up with and that current functionalities felt necessary but AU also reiterated the screen-reader compatibility being something that needed to improve.

3.3 Final Prototype

Some of the functionalities and how they work are presented in 2.3.2 but a full list of functionalities and accompanying details will be presented in this section and an image of the final prototype can be seen in figure 3.3.

3.3.1 Visual aspects

All the visual aspects can be seen in figure 3.3 and they are:

- Plotted Diagram - All 100 values are presented, each artist is presented on the x-axis in alphabetical order. On the y-axis the amount of listeners in the millions are displayed and in addition to position on the y-axis, the value is also represented by the size of the "dot".
- Toggle Box - The toggle box says "Toggle Diagram Description" and it also has a mouse-over tooltip saying the same thing. While toggled on it displays a white check-mark on blue background inside the box.
- Sliders - There are two sliders, the first one says "Start from data point:" and the value x can range from 1 to 100, default being 1. First slider also has a mouse-over tooltip saying "Select where to start listening from". Second slider says "Set volume" and the value x can range from 1 to 100 with default being 50. The second slider has a mouse-over tooltip saying "Set volume of sonification".

- Buttons - There are two buttons, the first is a blue button saying "Sonify Data" and the second button is orange and says "Stop Sonification".

3.3.2 Functionalities

The functionalities of the prototype will be presented in this subsection and Windows' Narrator was used to test screen-reader compatibility during the second usability test. The prototype can be navigated through by "tabbing" and then interacted with using different keyboard keys.

- The plotted diagram is an image and there was no screen-reader compatibility.
- The toggle box is toggle-able using mouse or pressing space while box has been selected and while "check-marked" the `accessible_output2` python package will narrate the pre-determined diagram description saying "The graph displays 100 artists in alphabetical order and their amount of monthly listeners ranging between 31.4 and 109.2 million listeners". The narration is played before the sonified data starts playing. Screen-reader narrates "Toggle Diagram Description" but does not recognize the tooltip.
- The first slider can either be dragged left or right using the mouse or by using the arrow keys, it can even be manually entered by selecting the number box to the right and then typing in the number preferred followed by "Enter". Typing letters in the number box does nothing after pressing Enter. The values on the slider can either be selected before starting the sonification or dynamically changed while the sonified data is already playing. The value on the slider decides where in the mp3 file that the sonification will start, selecting 8 makes sure that the first data point being played in the sonification is the eight value; if changed to 8 while the sonification is already playing it will simply change its position from where it is at to the eight value and then continue onwards. The screen-reader will narrate "Start from data point:" and also dynamically narrate which value is selected, once again the tooltip is not narrated.
- The second slider can be interacted with in the same ways as the first slider and the screen-reader compatibility is the same. The value of the slider dynamically decides the volume of the sonified data but does not affect the `accessible_output2` narration volume.
- The first button can be activated by clicking with the mouse or by pressing "Enter" or "Space" while it is selected. Activating the button will start the sonification and depending on the toggle box and the sliders it might play the narrator output before the sonified data and start at whatever position the first slider's value points at. The screen-reader will narrate "Sonify Data".
- The second button has the same interactability as the first one and screen-reader will narrate "Stop Sonification". Activating the button will stop the sonified data from playing, it is not a pause and it cannot stop the `accessible_output2` narration, it will only stop after it is done narrating.

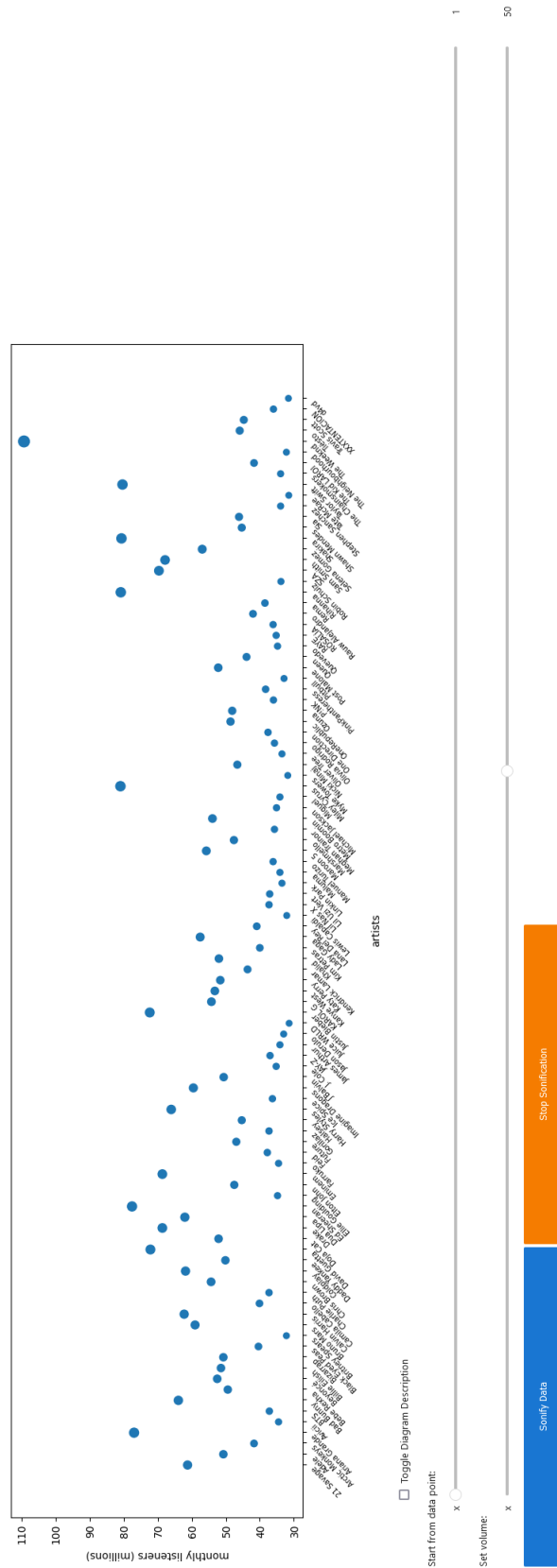


Figure 3.2: The figure displays a screen-shot of the entire final prototype without showing the JupyterLab UI

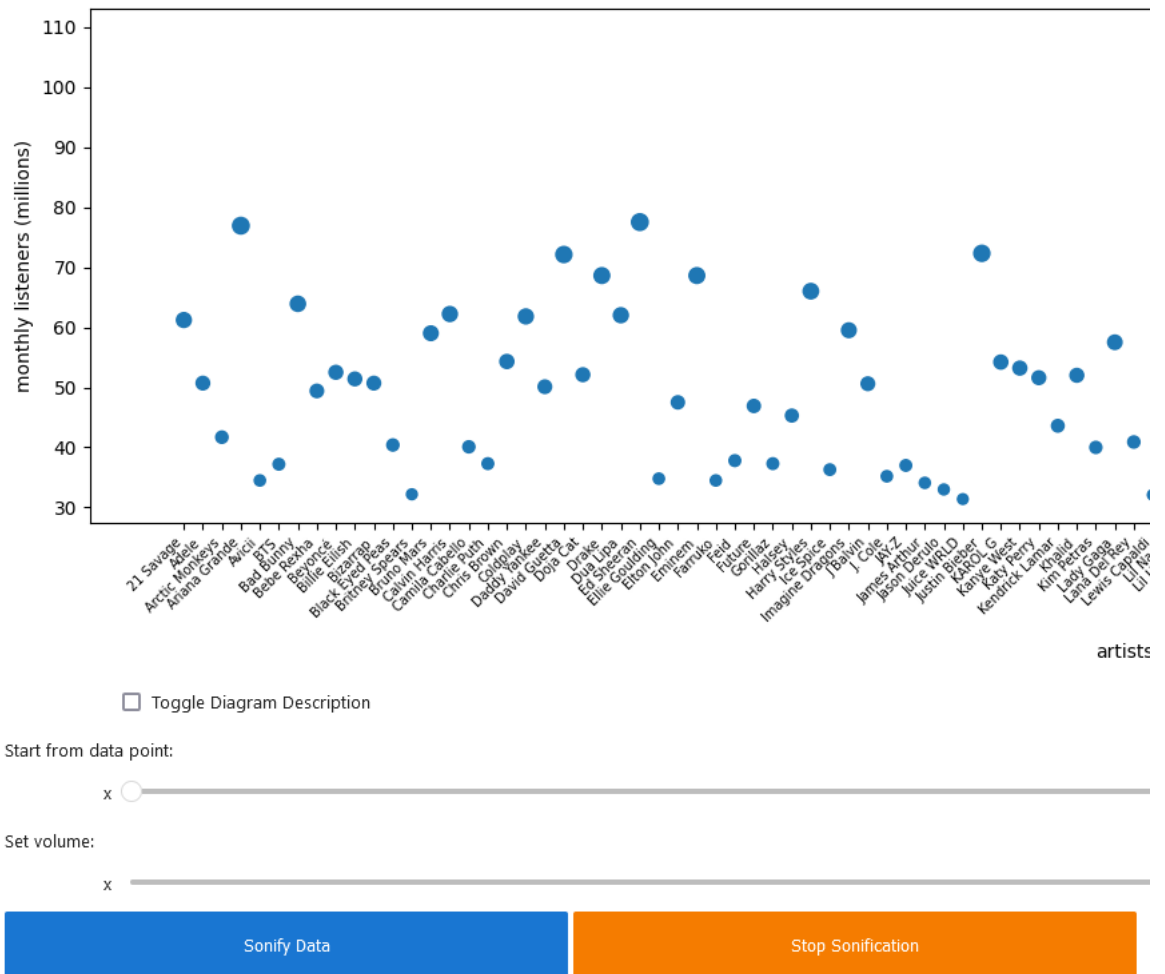


Figure 3.3: The figure displays a zoomed-in screen-shot of the final prototype without showing the JupyterLab UI

Chapter 4

Discussion

In this chapter the methodologies and the results are discussed. The results are interpreted with the methodology in mind, they are analysed in relation to the previous work and the research questions are answered. The prototype will be evaluated and its development discussed. Finally, some new questions and future work opportunities are discussed.

4.1 Usability tests and Tool Study

By asking questions about how the tests would be performed and what could be gained from the tests the planning of the tests became easier and it was a good way to do a individual brainstorm. Overall the resulting tests from the brainstorm questions fulfilled their objectives well but several flaws were discovered for both the first- and second usability test when performing the test with the imitated user, which only reinforces the decision to include an imitated test. When performing the tests with the actual user the tests looked a bit different due to the flaws being corrected which might be a double-edged sword, the test was better planned at that stage but the test-inconsistencies might translate to inconsistencies in the results that would not be there otherwise. The inconsistencies in the tests would be there even if no corrections were made, this due to the experience of using a screen-reader and navigating blindly were heavily favoured for the actual user compared to the imitated user; so introducing a few more differences in the test was a scenario where the benefits outweighed the detriments in this case. Because of the experience difference between the users the results of the imitated test would be deemed as less important compared to the actual test which is why making sure the actual test was as well planned as as possible was important. Performing the tests semi-structured was a positive because more feedback was provided due to encouragement or adding follow-up questions, it could however be seen as a lack in planning but in the end more planning might have limited the test participants' feedback.

The decision to perform the first usability test using an already existing tool was a good way to survey what kind of functionalities could be useful for a prototype but also to see how

the test participants interpret sonified data. In order to differentiate this thesis work it was necessary to see what might be missing from an already existing sonification tool, and by performing the first usability test with Desmos these "missing" features could be imagined. Choosing to test with Desmos instead of Sonification Sandbox was probably the correct decision. The interactability was quite limited seeing as the user could not listen to individual points or segments and with the overall difficulty of using the program it was a factor in the decision to use Desmos. The tool study should probably have been a bit more extensive since the Sonification Sandbox program seemed experimental and unfinished, finding one or two more tools would help in choosing an optimal tool to test on. But at the same time Desmos was a good choice and might have been picked anyway, so it is a question of time-management. In the end exploring more sonification tools would probably not be worth the time spent because existing tools were only used as inspiration for the prototype that would be developed.

Comparing this thesis test methodology to the test methodology of Siu et al. [6] lots of differences, but also some similarities could be discovered. The main difference between the two methodologies is that the purpose of Siu et al. was to compare two techniques of presenting information through sonification to screen-reader users while the purposes of this thesis are to evaluate auditory characteristics, software design, and interactability for people with visual impairment. Due to that main difference it is only natural that the test methodologies will differ. The amount of test participants are fewer in this thesis, two compared to sixteen. A higher amount of participants is generally advantageous but in this thesis the lower amount of participants is due to time constraints. The lower amount of test participants allowed for a more relaxed approach to the tests which gave the test participants room to explore and think through their answers which was reinforced by the face-to-face testing performed in this thesis while the tests made by Siu et al. were performed over Zoom. The tests were performed in similar ways, both in this thesis and in the tests by Siu et al. the general procedure was letting the test participants listen to things and then ask them questions to see how they interpreted the data in terms of trends or patterns. Both the Desmos test and the prototype test performed in this thesis were of the control representation i.e screen-reader descriptions were presented before the actual sonified data was playing which was one of the techniques compared in the tests performed by Siu et al. and it was the one they concluded was not as efficient as the audio data narrative representation. Perhaps introducing the audio data narrative into the prototype testing would have increased the user comprehension during testing, but the data was not that complex or long, and Siu et al. also concluded that the difference between the techniques was not as apparent when the data was less complex.

4.2 Test Results

In this section the results gained from the usability tests and the research questions will be discussed.

4.2.1 First Test

The results from the first usability test showed that the overall intuitiveness of sonification as a tool is quite good, it was a new concept for IU and they could still understand general

patterns of the data. Even though both test participants could understand the flow of the data there were some details that they could not answer, things such as the value of the data, exact shape, and the mathematical formula. After listening again with the assistance of a screen-reader both boundary values of the graph and the mathematical formula were narrated by the screen-reader and therefore acted as additional data that the sonification itself could not provide. Just like Siu et al. [6] argued that only descriptions were not enough, the argument found here is that the sonification itself is not enough either, so a combination of descriptions, read by screen-reader and sonification seems to be the way to go and is also the way chosen going into the second usability test. The two test participants answered that they preferred the description to be narrated before the sonification started a.k.a control representation [ibid], they were however not given the option to answer an audio data narrative representation due to the way the question was asked: "If you were to have both Sonification and a screen-reader, would you have the screen-reader automatically read before listening to the graph, after listening to the graph, or during the listening? Alternatively only on command?". But as discussed earlier the benefits of audio data narrative are not as important with simpler data sets, and the test included quite simple data sets making the inclusion of it as an alternative in the question a bit excessive. It might have been worthwhile to present it as an option and discuss with the test participants when it might be advantageous to see if they would agree to the conclusions made by Siu et al.

In order to follow a UCD process where the user is at the center of that process [8] some of the questions in the first usability test were asked for the sake of the prototype and the second usability test. Among the findings of these questions was the realization that non-continuous data was easier to interpret for the test participants which was also why the prototype has non-continuous data to listen to. The subject of sonification speed was another of those questions, both test participants thought that the $\frac{1}{2}x$ speed in Desmos was the one that was the easiest to interpret the graphs with. $\frac{1}{2}x$ speed in Desmos was quite slow which was one of the reasons for having the data in the prototype be slow as well. The third task in the first usability test was non-continuous and therefore easier to compare speed-wise with the prototype sonification, and AU believed there could have been a longer pause between the bars in that task which drove the prototype development towards an even slower sonification than the third task. The white noise in Desmos was something that confused the two test participants when the context of why it was there was not explained. After receiving information of its purpose they both thought that it was a good feature that added more information to the sonification showing that with proper learnability [ibid] it could be a beneficial functionality in the prototype.

4.2.2 Second Test

Quite similar to the results of the first test, the second test's sonified data could be interpreted pretty well by the two test participants, they could both hear the difference between the values and the general range. AU suggested that there should have been reference values to be heard beforehand so that you know whether a sound is high or low compared to all the other values, which could have been a conclusion made from the white noise feature of Desmos. The white noise acted as a reference value because a user could tell whether a value was below 0 or not and it should have been an indicator to include reference values in the prototype. The option to include the diagram description was received positively by

the test participants since they had both wished for screen-reading before the sonification in the first usability test once again showing that following the UCD process works well. The information provided by the description was also believed to be useful in understanding the sonification, but IU wanted it to be a bit slower and to round the values given to make it simpler. AU suggested including the reference values mentioned earlier in the description which would actually make a lot of sense because the user can toggle the description and the user might not want to hear the reference values every time they listen to the diagram. However having the reference values in the description would mean that when the user only want to hear the reference values the user would have to listen to the description of the diagram as well. The solution to both these issues could be to create another toggle box saying something like "Listen to Reference Values" which would only play the reference values and allow the user to pick and choose between both description and reference values being played before the sonified data.

In order to further evaluate whether the test participants could interpret the sonified data accurately the test asked them to describe the pattern of the last 10 values and after that compare the high- and low values with the first 20 values. The two test participants understood what they heard quite well in general, however the results suggest that IU and AU had different levels of perception and recall of the sonified data. IU's description was more general and lacked specific details, while AU's description was more comprehensive and included specific values. This does not necessarily mean that IU had a harder time remembering or understanding the values, it could be the nature of the question that just states to provide a pattern, and not specifics. It's important to note that the descriptions provided by IU and AU are subjective and may not match the actual pattern they heard. However, these results offer valuable insights into how different individuals perceive and recall auditory information. Future research can investigate the factors that contribute to individual differences in auditory perception and memory, and explore strategies for improving these skills. Comparing the last 10 with the first 20 in terms of high- and low values was easy for the two participants when it came to the highest value since it clearly stood out, it was in fact the highest value in the whole sonification. The lowest value was a bit harder because there were a few of them around the same pitch, and among the first 20 there was also more values close to that low value, despite all that both users could hear that it was among the lowest, and AU did mention that it was harder to do that comparison.

Both test participants had a positive opinion of the pitch assignment, they both thought it felt logical. Maybe that is because the sound is a piano sound and therefore familiar and pleasant, but the way that the MIDI note numbers were selected and assigned was meant to make it easy to follow which it seems to have accomplished quite well. The note duration received positive feedback as well, however, it should be something that can be changed, AU suggested that new users might want it to be slower while experienced users might want it to be faster, which is once again a byproduct of the prototype's learnability [8]. Being able to change the speed was something that was thought of before the test as well, but due to complications, see 4.3, it was never implemented in the prototype. The note velocity was also received well by the test participants, it was said to be logical and add realism to the piano sound but it was also noted by AU that it does not seem to add much to the sonification comprehension, and that the pitch is more important in that regard. While pitch arguably is the most important auditory characteristic when it comes to interpreting sonified data, the note velocity in the prototype's mappings is important because some values in the diagram

that are close to each other were assigned the same pitch, and therefore the note velocity was the distinctive factor. If the note velocity is proved to not bring any value, the pitch range could be increased and the necessity for note velocity as the distinctive factor would vanish. However, since it brought more realism to the piano sound it does in fact bring some value, whether that value is part of the comprehension or not is left to be explored. Overall the auditory characteristics seem to have been correctly configured but seeing as interpretation of the data at the beginning of the test was not perfect there is certainly room for improvement, tweaking the pitch would probably be the way to improve it even further, perhaps selecting MIDI note numbers with a larger range would be the way to proceed.

When asking the test participants about improving current functionalities or adding new ones the IU could not think of any improvements or missing functionalities, it could be that the prototype is well put together and accomplishes its purpose, but it could also be that the questions asked were too open-ended making it hard to think of an answer. It is probably the second reason in this case, seeing as AU had answers and that functionalities that were seen as important had to be scrapped during the prototype development. Besides the speed-changer functionality mentioned before, AU wanted better screen-reader compatibility, a 0-value divider if the data is continuous, and being able to choose instrument for the sonified data. The screen-reader compatibility being lacking is something discussed in 4.3, and more time should probably have been put into making it better, but the inexperience in using a screen-reader was something that limited the pre-test evaluation. The user interface was otherwise simple thanks to following the UCD principle and the guidelines given from the first usability test, see the end of 3.1. Seeing as the 0-value divider was seen as positive in the first usability test with Desmos it was only natural that AU would suggest including it in the prototype, provided that the prototype had continuous data presented, which it did not in this case. Adding an "instrument-chooser" which could be seen as an "for fun" functionality might actually add to the sonification comprehension and even if it does not, it still adds value in the same way that the note velocity does, by making the experience more pleasant.

4.2.3 Research Questions

How should the auditory characteristics be configured to enable smooth interpretation of sonified data?

During the second usability test it could be seen that pitch was probably the most impactful auditory characteristic, the range of pitches is the parameter that allows the users to interpret the value of different data points. The configuration of pitch was perceived well but there should definitely have been reference values to help the users' value comprehension, it was hard to tell whether a value was super high or especially super low compared to all the other values. So in short, the pitch should have a range big enough to differentiate between values, and an inclusion of reference pitch-values will help in interpreting whether a data point is high or low in comparison to the reference values.

The note duration was probably the second most important auditory characteristic when it came to interpretability, having the note duration at one second was seen as a good default value, it might even be a bit slower to make interpretation even easier as a new user. The realisation reached throughout the thesis work and through testing is that a note duration modifier is needed. New users would probably want it to be slower while more experienced

users want it to be faster, and being able to personalise the speed will help in interpreting sonified data. Summarily, a little bit longer than one second note duration as the default, with options to change the note duration to personal preference is most likely the way to go.

Finally the note velocity could be seen as the least important of the three auditory characteristics that were investigated, it seemed to only have brought a pleasantness to the sound of the data. It was seen as logical the way it was assigned to the data points so there are probably no changes needed there. The note velocity was not only important for sound pleasantness however, it also added more uniqueness to data points. There are only so many pitches to be assigned to data points, so by adding note velocity to data points it adds another dimension which enables bigger data sets to be used. A user can differentiate between two values with the same pitch if they have different note velocity, but it will most likely require some experience in listening to sonified data in order to interpret the difference successfully. So to sum things up, note velocity should not be disregarded when it comes to interpretation, and note velocity should be assigned like in this thesis, namely higher note velocity is given to higher pitch values.

How do you design a software prototype for someone with visual impairment?

Based on feedback and experiences from the two tests a few guidelines of what should be followed could be discovered. When designing a software prototype for people with visual impairment, there are two things that could be seen as the most important according to these discoveries. These two things are that it should be fully screen-reader compatible, and that it should be a minimalistic design. If elements in the prototype are not narrated by a screen-reader they are basically not accessible to the end user unless it is another audible element. Keeping a prototype minimalistic will also make sure that things like unnecessary text or visual design being narrated will not distract the user from what is important, which is usually the functionality of the prototype. Other factors important to the design are including section titles which help in localizing where in the prototype the user is at, and to ensure keyboard shortcuts are working properly with the popular screen-readers.

In what ways can the end user interact with the sonified data through the prototype?

In the end, there was a few ways for the end user to interact with the sonified data. First and foremost the end user could listen to the sonified data which was the most important feature of the prototype and one of the themes of this thesis. When listening to the sonified data the end user could also interact with it in certain ways, they could start and stop playing it, choose where in the sonification to start listening, decide if they want a description of the data beforehand, and change the volume of the sonified data. However, being able to change the speed of the sonification was also one way of interacting that was planned to be included initially, and as discussed earlier, it was also something that the end user felt was missing from the prototype. Perhaps letting the user interact with the positional slider in the prototype by using something tactile like a touch screen would increase the intuitiveness of the interaction.

4.3 Prototype

Initially, the idea of using a web-page as the prototype came from the pointers at the end of 3.1 saying that tabbing is usually how to navigate as a person with visual impairment and that screen-readers have two modes when navigating a HTML page. It was also the idea that a web-page could be kept minimalistic while also providing a user interface which most are comfortable with i.e the browser's search bar, toolbar etc. would not be distracting because users know they are there and what they accomplish. As mentioned, the web-page idea became very complex and knowing how well JupyterLab worked regarding the functionalities that needed to be implemented. It was an easy decision to choose the JupyterLab route especially considering the sonification video showing how to work in JupyterLab and what kind of python packages could be used for the prototype. Not to say there were no complications in using JupyterLab and python packages, during testing for example, several options, tabs, and JupyterLab functionalities unrelated to the prototype were interrupting the keyboard navigation. It was after the test was already done discovered that the prototype could be opened in a new window which would at least hide all the truncated cells in the JupyterLab file and therefore eliminate some of the interruptions. Overall the usability of the prototype was quite good and all the tasks of the second usability test could be accomplished without intervention. Looking at the six attributes of usability [8] and then the results of the second usability test it could be argued that the usefulness of the prototype was quite good. The efficiency and accessibility are tied together in this prototype, because of the navigation issues both of these attributes were lowered, however the prototype was developed with UCD in mind by following the feedback points received during the first usability test so the accessibility was not as low as the efficiency. If not for the JupyterLab interruptions, both of those attributes would be a lot higher.

Developing the prototype in JupyterLab was intuitive and fast paced, all changes could be seen visually in the same window as the coding was done in. Segmenting the code into different stages felt natural and once you were done with one stage of the code you could truncate it and focus on the next part knowing that the different segments were connected in terms of variables and functions. Changing the data set in the video guide from a two column numerical data set to a one column numerical data set added some complexity to developing the prototype, but it was also necessary in order to differentiate and to simplify the data; Russo's data both had more data points and were not spaced evenly on the x-axis, meaning that distinguishing individual values would be a lot harder than when they are evenly spaced as in the thesis prototype.

All the code led to a midi file being created and in order to fully use the prototype within the same program the midi file had to be converted into a mp3 file to play it within JupyterLab which was one of the limitations of the program. Lots of time was spent in trying to circumvent this JupyterLab limitation and in the end it did not work out. The decision was made to stop trying to fix it, even though having the data be in a midi file would allow for finding positional values within the file, which was only accomplished in the mp3 file due to every data point being one second long. Luckily, the one second long note duration was seen as positive for both users, but if it was seen as a bad value the whole slider solution for selecting a start value would need to change. It would mean that in order to keep trying to fix the midi-issue, the whole prototype would have to be redone from scratch; several of the python packages would have to be abandoned and the whole prototype structure would

change. Another change that was abandoned due to the same reasons as the midi-issue was the idea that after selecting a position it would play the sonification starting at that position and then play 10 notes before stopping. It was implementable, but at the cost of the stop function and the other dynamic functionalities not working while the 10 notes were playing. As presented in the prototype methodology some solutions were tried and in the end did not work. Probably the main functionality that was abandoned in the prototype was the ability to change the speed of the sonification, which was also something that AU said was missing from the prototype. Changing the speed of the sonification with the current implementation of the prototype would mean that it would have to be done manually, then converted into a mp3 file, followed by reloading the program and then finally playing it in the prototype and as discussed earlier, the positional slider would not work because every note would be faster than one second. Adding a toggle box added to the interactability and quality of life of the prototype and because it was easy to implement it was not really a big decision to make.

4.4 Future Work

Throughout the discussion there have been some future work possibilities that have been discovered, both in terms of prototype updates and potential research that can be expanded upon. Future updates to the prototype would include the note duration adjuster which would need some restructuring of the whole prototype to make it more dynamic i.e. changing speed would not require any manual work. Included with the dynamic restructure would be to use the MIDI file, and if that could not work, then at least not needing to manually convert the file into a mp3 file. Perhaps developing the prototype outside JupyterLab would help to facilitate a lot of these future updates, but then it would need to have a graphical user-interface (GUI) for it to make sense as a prototype or finished product. Making the prototype screen-reader compatible is another important future update that needs to be done, maybe with the new GUI it would gain that compatibility it needs. More future changes include the option to change instruments for the sonification, which could prove to increase interpretability for some users and provide some QoL to the prototype. Trying to connect the sliders with something tactile such as a touch screen or a touch pad might be interesting to implement and try out in the future.

When it comes to research questions and further research there are some things that would be interesting to investigate in the future. This thesis' research questions could be further analysed by implementing the prototype changes and then continue with usability testing to strengthen the findings of earlier tests. One could also with further usability tests and with more test participants continue to explore whether other pitch ranges would show a better sonification interpretation. It would also be interesting to investigate to which extent reference values would add to the interpretation of data and how detailed these references should be. If adding more test participants would show different answers than those found in this thesis, then the difference in answers could be investigated and evaluated.

Chapter 5

Conclusion

The purpose of this thesis was to develop a user-centered prototype for people with visual impairment, and to figure out appropriate auditory characteristics of sonification and how to efficiently combine it with screen-reader usage. This was achieved by usability tests and user feedback identifying pros- and cons of existing sonification tools and then translating those findings into a prototype that could be further analysed.

The findings suggests that sonification can translate visual information quite well, as long as context of the information is provided through a screen-reader, and that reference values are provided, the values in a visual graph can be aptly interpreted through sonification. Choosing a big range of pitches and adding another dimension in the shape of note velocity provides a sense of realism and differentiates values enough to where the difference in values can be heard through the sonification accurately. Having a long note duration as a default helps new users interpret sonified data and as users get more experienced they might want the note duration to be shorter for interpretability showing that a speed-modifier is a must-have functionality of a sonification tool.

Besides the speed-modifier, several ways of interacting with a sonification tool that are must-haves are being able to start- and stop the sonification, choose where in the data to listen, change the volume, and to ensure screen-reader compatibility. The screen-reader compatibility is also the main focus when developing a software prototype for people with visual impairment, keeping the prototype minimalistic by removing unnecessary elements, naming sections, and allowing keyboard- navigation and interaction are the main points to establish that compatibility.

The finding although convincing, are only of a small sample size and should probably be further evaluated by including more test participants. Future work could build on the findings of this thesis by implementing the suggested changes to the prototype and to explore how reference values should be used. Including more test participants could provide different findings, and those findings could then be compared to the findings of this thesis.

References

- [1] Thomas Hermann, Andy Hunt, and John G. Neuhoff. *The Sonification Handbook*. Logos Verlag, Berlin, 2011.
- [2] Nik Sawe, Chris Chafe, and Jeffrey Treviño. Using data sonification to overcome science literacy, numeracy, and visualization barriers in science communication. *Frontiers in Communication*, 5, 2020.
- [3] Nikki S. Rickard, Samia R. Toukhsati, and Simone E. Field. The effect of music on cognitive performance: Insight from neurobiological and animal studies. *Behavioral and Cognitive Neuroscience Reviews*, 4(4):235–261, 2005. PMID: 16585799.
- [4] American Foundation for the Blind. Screen readers. <https://www.afb.org/blindness-and-low-vision/using-technology/assistive-technology-products/screen-readers>, 2023. [Online; accessed 26-April-2023].
- [5] World Wide Web Consortium. Web content accessibility guidelines (wcag) 2.1. <https://www.w3.org/TR/WCAG21/>, 2018. [Online; accessed 26-April-2023].
- [6] Alexa Siu, Gene S-H Kim, Sile O’Modhrain, and Sean Follmer. Supporting accessible data visualization through audio data narratives. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI ’22, New York, NY, USA, 2022. Association for Computing Machinery.
- [7] ECLKC. Using a data narrative to tell a story. <https://eclkc.ohs.acf.hhs.gov/publication/using-data-narrative-tell-story>, 2021. [Online; accessed 26-April-2023].
- [8] Jeffrey. Rubin and Dana. Chisnell. *Handbook of usability testing : how to plan, design, and conduct effective tests*. Wiley Publ., Indianapolis, Ind., 2008.
- [9] Ramiro Velázquez. *Wearable Assistive Devices for the Blind*, pages 331–349. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

-
- [10] Sergio Mascetti, Lorenzo Picinali, Andrea Gerino, Dragan Ahmetovic, and Cristian Bernareggi. Sonification of guidance data during road crossing for people with visual impairments or blindness. *International Journal of Human-Computer Studies*, 85:16–26, 2016. Data Sonification and Sound Design in Interactive Systems.
- [11] Franco Delogu, Massimiliano Palmiero, Stefano Federici, Catherine Plaisant, Haixia Zhao, and Olivetti Belardinelli. Non-visual exploration of geographic maps: Does sonification help? *Disability and Rehabilitation: Assistive Technology*, 5(3):164–174, 2010. PMID: 20415530.
- [12] Sharona T Levy and Orly Lahav. Enabling people who are blind to experience science inquiry learning through sound-based mediation. *Journal of Computer Assisted Learning*, 28(6):499–513, 2012.
- [13] R Weir, B Sizemore, H Henderson, S Chakraborty, and J Lazar. Development and evaluation of sonified weather maps for blind users. In *Designing inclusive systems: Designing inclusion for real-world applications*, pages 75–84. Springer, 2012.
- [14] Ather Sharif, Olivia H. Wang, and Alida T. Muongchan. “what makes sonification user-friendly?” exploring usability and user-friendliness of sonified responses. In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS '22*, New York, NY, USA, 2022. Association for Computing Machinery.
- [15] R. Wang, Christoph Jung, and Y. Kim. Seeing through sounds: Mapping auditory dimensions to data and charts for people with visual impairments. *Computer Graphics Forum*, 41, 2022.
- [16] Project Jupyter. Jupyterlab is ready for users. <https://blog.jupyter.org/jupyterlab-is-ready-for-users-5a6f039b8906>, 2018. [Online; accessed 24-March-2023].
- [17] Wikipedia. Euclidean distance — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Euclidean%20distance&oldid=1136196593>, 2023. [Online; accessed 01-February-2023].

Appendices

Appendix A

Test-Interview Answers

A.1 Initial Usability Test

Can you tell what you are listening to?

Imitated user:

1. Think the graph is exponential but hears that it is increasing in value over time.
2. Hears that the value is increasing, then flattening and then increasing again.
3. Hears that the bars are decreasing in height early on followed by a big bar that increases and then decreases. Last bar is of the same height as the third one (right answer is second one) and the user can tell that there are pauses between some bars.

Actual user:

1. Rising value, sound changes in the middle.
2. Rising in value at the start then the rise stops and then continues to rise again.
3. Could sense it was a histogram without being told it is. First three bars are decreasing in value, 4th one is probably 0, three more values and the one in the middle of those three values was the highest and also the highest overall.

What changes would you make to the sound in order to make it easier to understand? (user is told what the graph is before the question is asked)

Imitated user:

1. Lower pitch as a start value would make it easier to interpret, the static sound below x-axis makes it harder to understand without knowing what it means (meaning not explained yet).
2. Same as last one but it was easier to interpret due to increased focus compared to the last task.
3. User thinks it might be easier to understand if the sound was one octave higher or if the sound was more relatable such as piano sound especially because the sound is not continuous like the first two tasks. User added that first two tasks would be easier to interpret if they were non-continuous sounds.

Actual user:

1. Good that the speed stays the same when the slope stays the same. The static sound below x-axis makes it harder to tell that the graph is not changing in the middle without its context.
2. After being told what the graph is, the sound makes a lot of sense.
3. With a little bit of training and some reference values it would be easier to interpret but you could tell how the bars looked in comparison to the other bars.

What speed do you prefer when it comes to understanding what you are listening to?

Imitated user:

User thinks that $\frac{1}{2}x$ speed was the easiest one to understand because the "flat" sound was easier to pinpoint at that speed.

Actual user:

$\frac{1}{4}x$ speed would be good if the graph is very complex, acceleration is easier to interpret but it is not as interesting to listen to at slower speed when you want the "full picture". $\frac{1}{2}x$ speed is good for both seeing the full picture and interpreting the graph, it was better than $1x$ and should have been the default. Both $2x$ - and $4x$ speed could be usable for a longer curve but it did not give any value to the test graph (which was quite short). In short: slower speeds

are good when looking at sections of a graph (short curve) and higher speeds are good when you want to see the full picture for a long curve.

Can you tell if a point you are listening to is below the x-axis or not?

Imitated user:

Yes because of the low pitch.

Actual user:

It is the static sound, it is smart to have as long as you understand why it is there before using it. The static sound proved to be a bit disruptive without that knowledge.

What information are you missing from the reader i.e what would you like it to add?

Imitated user:

The information given was sufficient.

Actual user:

The information is sufficient but the navigation and interface of the information was difficult in Desmos.

Does the screen-reader provide information you did not receive by listening to the graph?

Imitated user:

Yes you heard the exact formula ($5x^3$) and the boundary values of the different axes.

Actual user:

Boundary values cannot be interpreted from sonification only, you can only tell if the values are below the x-axis or not.

If you were to have both Sonification and a screen-reader, would you have the screen-reader automatically read before listening to the graph, after listening to the graph, or during the listening? Alternatively only on command?

Imitated user:

Automatically before listening to the graph which provides information that lets you prepare and focus on the listening part. You cannot really concentrate on both sonification and screen-reader at the same time. If screen-reader would read information after listening to the sonification you would want to listen to the graph again after the screen-reader. Clicking a button could work but auto-playing would be more appropriate.

Actual user:

Boundary values to be read, short summary of graph and then followed by the sonification. One alternative option could be a voice saying something at

the same time at slower sonification speeds but this option should not be a priority although it could be nice to have. Keyboard shortcut to activate the screen-reader graph information would be nice in addition to the automatic one.

Would you like to see more information coming from the screen-reader and less from sonification, the split it has now or the other way around?

Imitated user:

They complement each other well and should not be used on their own. The dynamic between them should not be changed, no more or less information from either part.

Actual user:

The balance between them seem healthy, sonification could have different tones for different colors in the bar diagram case though.

A.2 Second Usability Test

Did you get a good understanding of what you're listening to? Did the diagram explanation give you enough information?

Imitated user:

Got a decent understanding of the range of values and how they were “moving” from one point to another. Diagram explanation was a bit fast, and the numbers should probably be rounded because they made it a bit confusing.

Actual user:

Good idea with pleasing sound, can also speed it up without it being a problem. can hear the difference in sound from different points, but a reference value to constitute what's a high value and what's a low value would be needed. The reader gave useful information, the end points were good, but could include the reference points in the diagram description.

What is the pattern of the last 10 notes?

Imitated user:

Starts at low pitch then moves in a zig-zag pattern and stops at an even lower pitch than the start.

Actual user:

First one was low, one value was pretty high, among the last ones there were some that were the same and one that was very low. The highest value was close to the end.

Were the lowest and highest values among the 10 you just listened to high

and low respectively compared to the first 20 values?

Imitated user:

Yes they were.

Actual user:

The highest value was for sure higher than the values from the first 20 but the lowest among the 10 was harder to compare with the first 20.

Do you think x was correctly assigned to the values? x = pitch, note duration and note velocity. If not, how would you change it?

Imitated user:

Pitch was assigned logically. Note duration was good. Note velocity was also logical.

Actual user:

- **Pitch:** With points, bars and things like that it is very logical the way it is now, with a graph the sound should probably be continuous.
- **Note Duration:** Was a good standard the way it was, with more experience listening it could be increased as well. For “new” ears even slower might be preferable but after listening a couple of times the speed seemed healthy enough. There should be an option to change the speed of the sonification.
- **Note Velocity:** It sounds good to the ears since it’s like a real piano but for apprehension it might not make a difference no matter how you set the note velocity. Pitch is of more importance when it comes to understanding the sonified response.

Is there any functionality missing from the prototype that you know would enhance the usability of it?

Imitated user:

Could not think of anything

Actual user:

Make it more screen-reader compatible, it does not read everything automatically (might be windows screen-reader limit). If using graph data having an origo divider would be nice and making it continuous. Fun to have: being able to switch between different sounds (piano, guitar, drums etc.). Piano sounds nice and it is easy to hear high and low pitches when it’s a piano, but might not be preferable for everyone. Being able to change the speed of the sonification would also be nice because as you gain more experience you want it to be faster and if you don’t understand what you’re listening to you might

want to slow it down.

Were there any existing functionalities you felt were unnecessary or who needed changing?

Imitated user:

No, everything felt necessary and worked well.

Actual user:

No it would only be the screen-reader compatibility in that case. Add key-binds for the different buttons and sliders. Simple interface where it's hard to mess up.

Appendix B

Prototype

Link to github: <https://github.com/JackeJacob/Exjobb-jupyterlab>, remember to read README for the packages that are needed to run the program.