Accessible Weather Forecast Visualization

Abigail Albuquerque, Kaley Du, Justin Weintraub, Aria Yan

1 Introduction

An estimated 15% of the world's population experiences some form of disability. They face barriers in regular life, including access to information and communication technology [4]. Weather is a very commonly visualized type of data and weather apps are some of the most commonly used across the globe. The data is usually visualized in the form of global maps, charts, heatplots, etc [6].

We aim to address this issue by evaluating existing weather apps and developing a more accessible weather visualization tool. Drawing from recent research, our goal is to create a platform that offers comprehensive weather information while being inclusive to users with disabilities.

Our primary focus is on informing a user regarding the weather across locations in the US . We aim to provide information on various aspects such as temperature, weather conditions, wind speed, etc. This will be achieved through a combination of accessible visual and sound-based techniques, offering users a multi-sensory experience.

Additionally, we will learn to work with live API calls, and learn more about how to create accessible visualizations and combine audio and visual components for the same. We hope to bridge the gap between weather visualization and accessibility.

Analyzing apps using Chartability's framework, we found that the most common problem with accessibility tools was that finding information regarding the weather simply through accessibility tools like VoiceOver functionality was extremely tedious. Further, despite being tech-savvy, we found it very hard to navigate the functionality. We realized that a lot of visual components of the apps are being read unnecessarily as well. Through our efforts, we hope to improve this design and make finding information simpler and less tedious.

There are a lot of aspects of the Chartibility framework that we plan to consider for our final design. Chartibility's tests are split under the following categories: perceivable, operable, understandable, robust, compromising, assistive, and flexible. For perceiving, we need to make elements visually distinct and easy to see. For operability, we're have to make sure our app is usable through screen readers, by low vision people, and having the apps operatable by methods such as text to speech and keyboard inputs. For understandibility, we need to make sure there's a lot of informative explanations. We need to ensure our app runs smoothly for the user for robustness and assistiveness. For the compromising category, we need to consider adding a table for the data in addition to the sound/background based visualizations. And for flexibility, we have to make sure our app works for different users and for different screen sizes.

2 One-sentence description

Our project aims to create an accessible weather visualization, improving on current techniques used by popular weather apps, using live API calls and innovative visual and sound-based techniques.

3 Project Type

Application (Live API)

4 Audience

The primary audience for this application are low-vision and blind users, as well as other disabled users who can benefit from accessible features. While other apps exist that help accommodate these demographics, we are hoping to combine elements from existing research and frameworks as well as incorporate sound

to create a new app that is usable for everyday people and is accessible as possible.

5 Approach

> What is your approach and why do you think it's cool and will be successful?

At the top of the page, we will present the user with a map of the US. The user will be able to select a location in the US. We also plan on implementing a speech-to-text option, allowing users with need of accessibility input a location easily. A toggle button asking users if they are visually impaired will be shown on the same page. This toggle will turn on the accessibility settings on the page to have high-contrast, enable automatic screen reading/summarization tools, etc. We aim to have all of these functionalities work with a screen reader as well. After a location is selected, a new page will display the weather in the area for approximately the next 3 days.

The page with the weather will have several features including a back button, a box showing the weather details, a weather alert box, an artistic visualization of the weather combined with sound effects, a time/animation slider at the bottom of the page that allows to play/pause the animation of the weather for the next 3 days. We also plan on having a button allowing users to view the data in tabular format.

Our app is different than other weather based applications due to it's focus on accessibility with regard to the Chartability framework, which is a newer framework. Our experience will hopefully allow disabled and blind or low-vision users to use the app easily.

6 Best-case Impact Statement

> In the best-case scenario, what would be the impact statement (conclusion statement) for this project?

We create an accessible weather experience through auditory and visual elements.

7 Major Milestones

Here are a few of our major milestones:

- Research existing frameworks and best practices for sound and text to speech

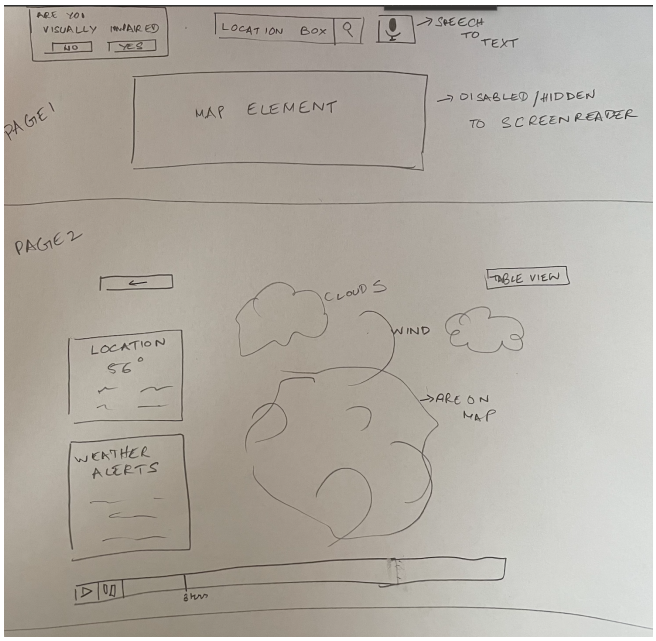

Figure 1: Paper sketch of our design.

- Decide on compliance audit framework
- Integrating the map component into our project
- Determine and implement appropriate API
- Find the sound and text to speech components to be used
- Evaluation and full integration of Chartability critical components
- Integrate additional accessibility components (if time permits)

8 Obstacles

8.1 Major obstacles

Among the 14 critical heuristics laid out by Chartability, some of them are much harder to fix than others. Beyond these heuristic based obstacles, we also have some functionality obstacles.

- Heuristic: Content is only visual

  Screenreaders such as JAWS or VoiceOver must be able to access all chart/displayed information on the page. We also need to synchronize any audio description with any potential animations that we have. To do this, we need to think carefully about how we structure and build our visualizations, so that we design with intention.

- Heuristic: Controls override AT controls

  We need to make sure that custom keyboard and touch controls do not override screen reader settings. Since we will be definitely be implementing many different controls for input and page navigation, we need to be careful that these controls can be turned off and remapped.

- Finding a suitable API

  For our app to work, we need to find and be able to use a free API that can get weather data at least for the last 72 hours. If we can't find an API that meets our needs, we'd have to rework our app to adjust for API limitations.

- Real time API updating

  Our app will be pulling from the API on load and on user events. We have to make sure we are able to perform API requests in real time, else, our application won't work as we can't get weather for specific locations.

- Integration of map coordinates

  A major component of our application is being able to get map coordinates from location, which is then passed on to the API. This component is crucial as we won't be able to make API calls for the weather of a location without the location coordinates.

8.2 Minor obstacles

- Designing tab stops

  Every interactive element (that represent buttons, links, or selectable features) must have a tab stop, while non-interactive elements should not have a tab stop.

- Technological support

  We need to make sure there are diverse technological means to access the chart and its information and functionality. This means we need to test our tool with as many different devices, browsers, and other tools as possible.

- Other Critical Chartability Heuristics

  Here are a set of other heuristics that we need to make sure our app avoids: Low contrast; small text size; seizure risk; interactive modality only has one input type; no interaction cues or instructions; no explanation for purpose or for how to read; no title, summary, or caption; reading level inappropriate; no table; data density is inappropriate; navigation and interaction is tedious; and user style change is not respected.

- Allowing vision impaired people to use the app

  One challenge is making it so the app is accessible enough so that a blind person can traverse and use it without additional support. This will require a lot of research.

- Working with the Maps API

  It will take some time to understand how the Google Maps API works, and how we can create events for clicking on specific areas. If we don't succeed at this task we can forego the map component and find another method of selecting areas like a dropdown list.

9 Resources Needed

What additional resources do you need to complete this project?

We plan on using data from the weather.gov API. One API call can return up to 168 hours of hourly forecasts. We plan on using 72 hours of forecasts from this API to create our visualization. Most of the data cleanup performed will be to classify forecasts such as "Mostly Sunny" and "Partly Cloudy." We will also have to parse through the textual windspeed data, as some data points include a range ("13 to 15 mph") and some don't ("25 mph"). Fortunately, the data values for temperature, probability of precipitation, dew point, and relative humidity are all numerical. One reason we're choosing to use the weather.gov API is because we were interested in working with live data that changes and updates every day in addition to the API being free.

For the audio component, we plan on using applications like Soundcloud and Youtube to find and download the audio that satisfies our needs. We can also create our own audio clips. To highlight intensity of certain weather conditions, we plan on playing with the sound and speed components, in addition to overlaying multiple audio clips.

We plan on using a text-to-speech api to help implement such a feature, and other apis that will allow our cite to comply to the chartability framework.

For the visual component, we plan on using d3.js to generate any visualizations, for example the clouds, rain, etc. We plan on using other visualization aspects like number sliders, etc through d3.js as well.

10 Related Publications

With regards to accessibility, the paper "Designing Accessible Visualizations: The Case of Designing a Weather Map for Blind Users" by Dustin Carroll, Suranjan Chakraborty and Jonathan Lazar [1] will help us determine how to make our visualization more accessible. It especially can help tailor our app to blind and vision impaired users.

The paper "Visualization in Meteorology—A Survey of Techniques and Tools for Data Analysis Tasks" [5] by Rautenhaus et al. provides an overview of existing visualization techniques. This will provide us with a guide for best practices when it comes to visualizing weather. It is also a starting point for understanding various other challenges.

The paper "How accessible is my visualization? Evaluating visualization accessibility with Chartability" by Frank Elavsky, Cynthia Bennett, Dominik Moritz [2] provides an accessibilty evaluation framework known as chartability. This is a framework we can use to evaluate the accessibility of our weather app. This framework includes 7 main principles: Percievable, Operable, Understandable and Robust, from the preexisting POUR principles, in addition to Compromising, Assistive, and Flexible. We will try to design our visualization around this framework so that all of the critical requirements are met.

The paper "Accessible Visualization: Design Space, Opportunities, and Challenges" by N. W. Kim, S. C. Joyner, A. Riegelhuth, Y. Kim [3] includes an analysis of papers published in the last 20 years before this paper was published regarding accessilbility in visualization. The authors mapped a design space for accessibility that includes: user group, literacy task, chart type, interaction, information granularity, sensory modality, assistive technology. They also discuss current knowledge gaps, opportunities, and challenges. This will be a good overview that helps us get an empirical view of what is going on in the accessiblity visualization space. It's framework will be useful as Chartability mentions that it should not be used in place of a compliance audit but alongside it.

The paper "Development and Evaluation of Sonified Weather Maps for Blind Users" by Weir et al.[7] describes methods to sonify weather information for blind users. It draws from the Web Content Accessibility Guidelines (WCAG) from the Web Accessiblity Initiative. This will be useful as an existing example of using sound to make weather visualizations accessible. Though it may be entirely auditory, we can take elements from it and apply it to the Web Content Accessibility Guidelines along with the Chartability framework in our own project.

11 Define Success

> When / How do you know if you have succeeded in this project?

We will succeed when we have a weather app that shows at least temperature and whether it is rainy/sunny/cloudy for the next three days in a way that marks all of the critical requirements for the Chartability accessiblity framework.

References

[1] Dustin Carroll, Suranjan Chakraborty, and Jonathan Lazar. Designing accessible visualizations: The case of designing a weather map for blind users. volume 8009, pages 436–445, 07 2013.

[2] F. Elavsky, C. Bennett, and D. Moritz. How accessible is my visualization? evaluating visualization accessibility with chartability. *Computer Graphics Forum*, 41(3):57–70, 2022.

[3] N. W. Kim, S. C. Joyner, and Y. Kim A. Riegelhuth. Accessible visualization: Design space, opportunities, and challenges. *Computer Graphics Forum*, 40(3):173–188, 2021.

[4] World Bank & World Health Organization. World report on disability: Main report, 2011.

[5] Marc Rautenhaus, Michael Böttinger, Stephan Siemen, Robert Hoffman, Robert M. Kirby, Mahsa Mirzargar, Niklas Röber, and Rüdiger Westermann. Visualization in meteorology—a survey of techniques and tools for data analysis tasks. *IEEE Transactions on Visualization and Computer Graphics*, 24(12):3268–3296, 2018.

[6] K. Riley, D. Ebert, C. Hansen, and J. Levit. Visually accurate multi-field weather visualization. In *IEEE Visualization, 2003. VIS 2003.*, pages 279–286, 2003.

[7] R. Weir, B. Sizemore, H. Henderson, S. Chakraborty, and J. Lazar. Development and evaluation of sonified weather maps for blind users. In Patrick Langdon, John Clarkson, Peter Robinson, Jonathan Lazar, and Ann Heylighen, editors, *Designing Inclusive Systems*, pages 75–84, London, 2012. Springer London.