CS235 Project, Spring 2017

# NextStop

*Gets you there!*

An ultra-accessible web-app intended to be used by the elderly and other people with limited vision (color blindness, partial blindness, etc). The app would read VTA and CalTrain timetable information and provide it to the user in an accessible manner using novel accessibility techniques along with auditory feedback.


Team: Aesthetik Designs

- Avinash More
- Krithika Raghavendran
- Nicolas Mauthes
- Vikram Deshmukh
- Vishrut Sharma

# Table of Contents

## Introduction

An ultra-accessible web-app intended to be used by the elderly and other people with limited vision (color blindness, partial blindness, etc). The app would read local transit timetable information and provide it to the user in an accessible manner using intuitive colors as well as auditory feedback. It will also allow the user to plan trips ahead of time and mark certain routes as favorite for easier access.

## Competitor analysis

The market is flooded with solutions that aim to provide similar services and features. Some of the most famous ones are Google Maps, Transit App, Apple Maps, etc. While the popularity of these apps oscillates between two ends of the spectrum, none of these provide features that would cater to our target user base. This is where the novelty of our app comes in. We will provide information to our users, in the most convenient and accessible way possible. The UX of the existing products is too tedious if not complicated for laymen users to grasp and use optimally. We have planned a disruptive UX for our product that will help us set our app apart from the competition.

## Target user base

The project intends to serve the needs of the generally overlooked user base when it comes to tech- the elderly and people with limited vision including people with partial blindness and color blindness who prefer travelling by public transport. Our app will be distributed as a web app, out-of-the-box, with no installation or sign in necessary. Being a web app, it will be platform independent too. We intend to open source the app and also *not* monetize it. We expect the community to then take on the mantle of developing the app further and maintaining it.

## Task analysis

Clearly the most critical task that the app performs is its ability to find transit information for the user. This will at first involve the user entering his or her location and a new destination as well as the preferred departure or arrival time into the app. The user will then be able to view a list of transit options based on what was previously entered. As the user spends time with the app and saves more routes to Favorites, this process becomes much faster, and selecting a route from Favorites will be preferred when the user is not heading to a new destination. The ability to toggle a colorblind-friendly color scheme is an important of feature of the app, and we expect that colorblind users will find this feature useful.

## Features

The product includes many novel (and some semi-novel) features. Some of which are listed below:

- Text-to-speech
    - Every time the user taps the screen, the text underneath is read out by a TTS engine, be it a button label or any other information text. We are planning to use the TTS engine that is shipped with the Chrome Browser. This will enable the app to be bandwidth efficient by not making an API call to convert text to speech every time the user interacts with the app.

- Colorblind friendly mode
    - The app features a high-contrast (i.e. black and white) mode that will make it easier for users with colorblindness to differentiate between various aspects of the app.

- Bigger Buttons, larger font
    - The GUI sports big, block buttons that are easier to interact with. The UI is designed to be uncluttered and have only the most important controls required to use the app. Additionally, the font sizes are quite large as compared to the font sizes in traditional apps in order to make it easier to read for people with weaker eye-sight.

- Favorite Routes option
    - To save time for the user, we provide an option to add a route as favorite. This enables the user to get access to frequently accessed routes quickly. This can prove to be extremely useful in case of an emergency and for people with memory-related medical conditions like dementia, etc.

- Take Me Home! (Version 2.0 feature)
    - The app will eventually contain a feature that will listen to voice commands from the user. The user would be required to speak the phrase ‘take me home’ while the app is running. And then in case of an emergency, the user may just say ‘take me home’, the app would relay navigational instructions via voice to the user. **This feature is still under review.**

- Platform Agnostic!
    - It will be an HTML5 based web app (thereby being completely platform agnostic). Therefore, no installation required: Just visit the app once and add the icon to home screen.

- Free for All!
    - It will be free for all users. No registration/login will be necessary.

## Experimenting with Accessibility

With an app like this one, novel solutions to enhance accessibility were needed beyond just following the standard accessibility guidelines. Summarized below are the approaches that we undertook to further the level of accessibility of our app.

### Approach 1: High Contrast color scheme

Well before our first meeting, during the paper-prototyping phase first few ideas revolved around having a dual-mode for our app:

- Normal mode for regular users
- High-contrast mode for users with vision disabilities

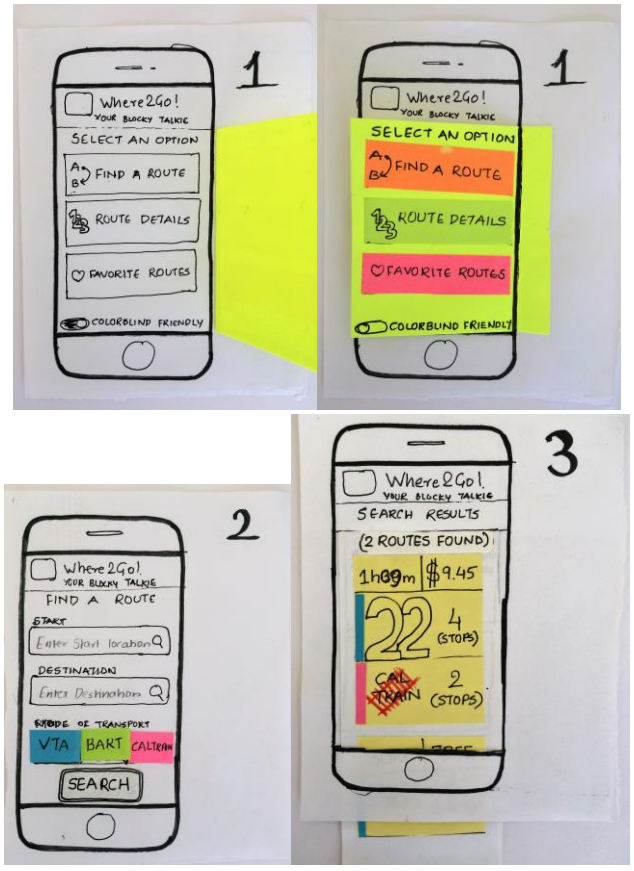



Fig.1. Paper prototype

These design considerations were not necessarily motivated by strict accessibility guidelines and in fact prove counter-intuitive when we look back because we ultimately chose colorblind users as our target audience who would not necessarily benefit from anything we deem as '*normal*'.

### Approach 2: Types of Colorblindness and using 3 different color palettes

When we had our initial few meetings and sat together to brainstorm and put ideas on paper, we decided to narrow down our target users to those with colorblindness since we could make the focus and the USP of our app more concrete. We started out really ambitious by having 3 different color schemes for the app corresponding to the 3 different types of colorblindness that persists, namely:

- Deuteranopia
- Protanopia
- Tritanopia

We had chosen to go with Atomic as our prototyping tool and hence decided to parameterize the different color schemes chosen into 3 options in the settings panel which would allow the user to choose whichever mode he/she felt comfortable with.

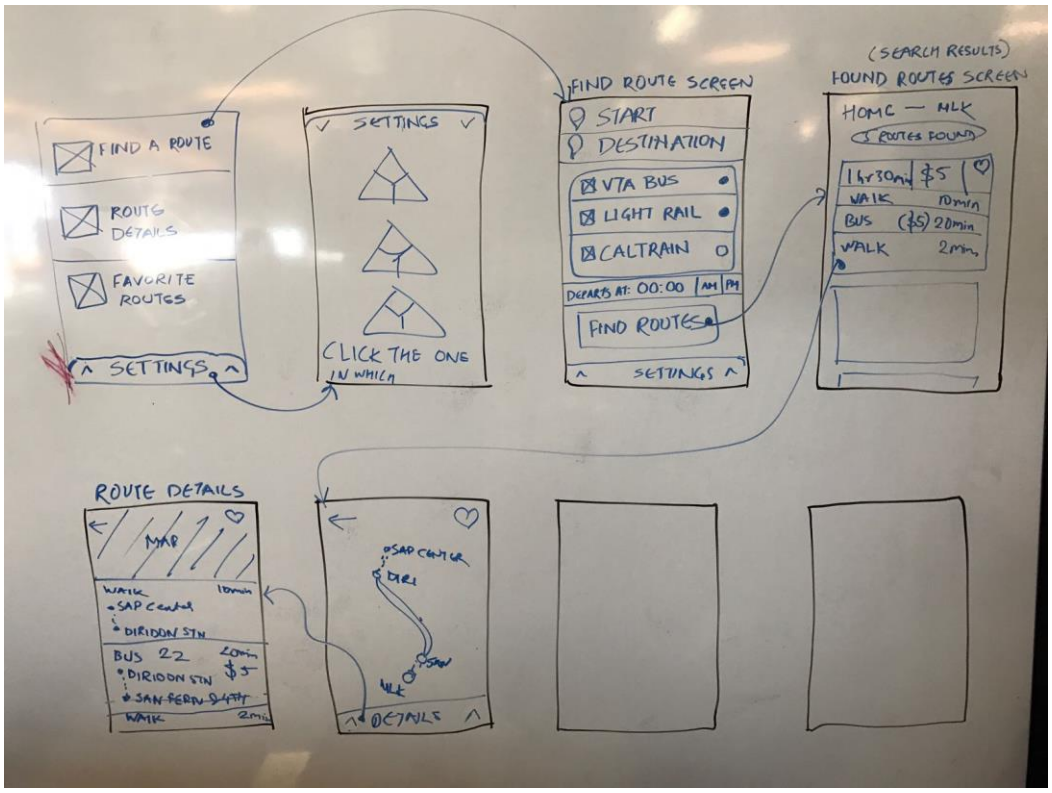



Fig. 2. Wireframes

Each color mode would have 3 types of colors that were shaded into parts of a triangle and we used a tool called *ColorBrewer (http://colorbrewer2.org/#type=diverging&scheme=RdYlBu&n=4)* to figure out the different color schemes that would be suitable to the different variants of colorblindness. These 3 modes were discrete from the 'normal' mode available to regular users which had a color scheme that included pastel shades since they are visually pleasing. The Settings bar would be present on every screen and could be brought up at will whenever the user so pleased.

### Approach 3: Combination of Grayscale Color Scheme and Pattern fill

After the first round of usability testing, we decided to get rid of the Settings panel altogether based on the feedback received that it proved to be obtrusive and also based on how often it gets clicked on (which was not that often at all!).

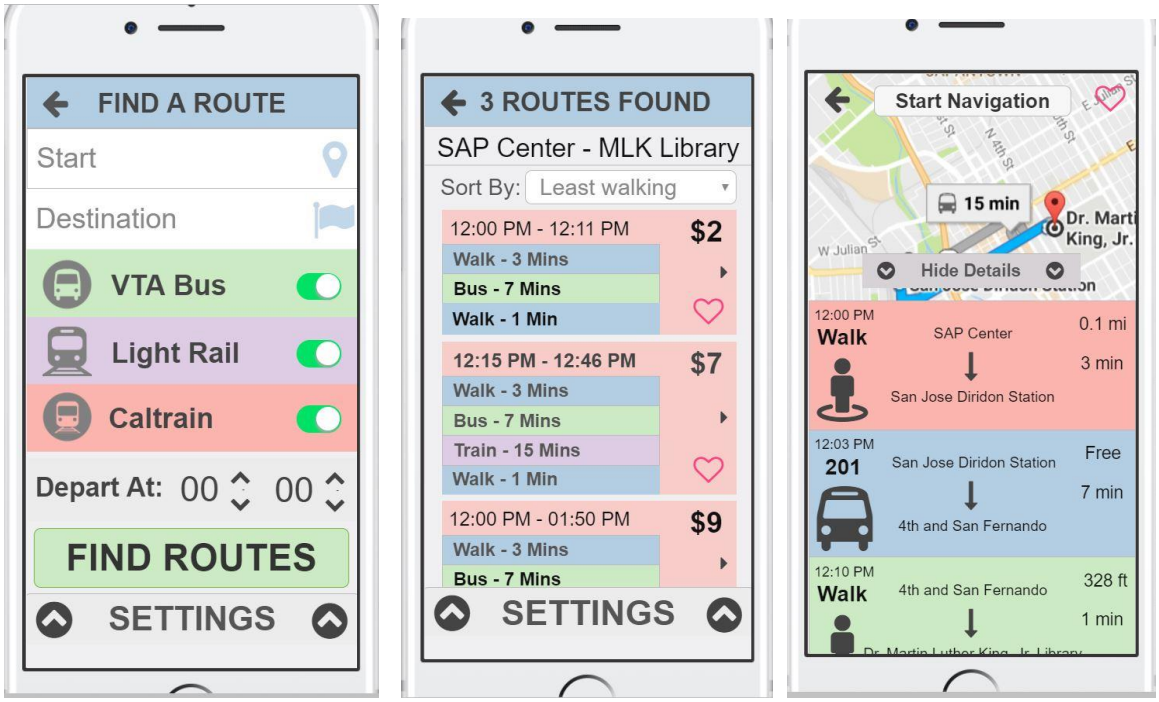



Fig.4. App with color scheme

By this point, we had already made the switch over to a different prototyping tool (Mockingbot) which did not offer the parameterizing option that was present in Atomic. So our design choices at this point were to *brute-force* it and manually re-create each screen 4 times for the four different color schemes we had decided to go with. This proved too tedious, and thus was born the final design choice of going with a classic choice of Black & White.

We got this suggestion from one of the senior members of the SJSU UXA group with whom Vikram had spoken to in order to get some meaningful feedback. We also decided to implement pattern-fill in order to give the tiles some additional granularity and visual separation. On the map, we added appropriate symbols for ‘walking’ and ‘bus’ along the trails in order to clearly demarcate areas that required walking and taking bus/rail.

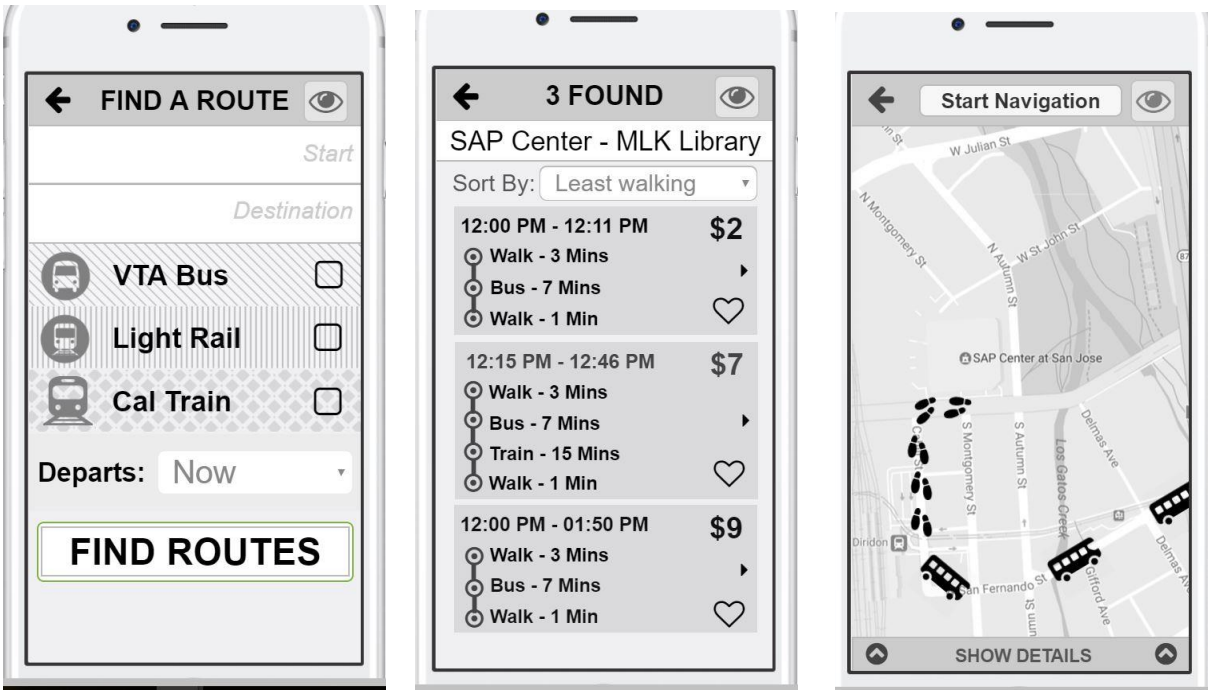



Fig. 5. App with grayscale

## User Testing

We had learned the importance of user testing in our class. We had more than an hour of discussion on topic. We discussed following things related to testing.

1. What are the types of user testing?
2. How to conduct user testing?
3. How many users are optimum number of users?
4. How many number of people are required to conduct the testing?
5. What are the roles and responsibilities of each person involved in conducting testing?

Along with the answers to these questions, we all followed all the guidelines we had discussed in the class. For user testing, we had decided the testing task as follow:

> *"Find the cheapest route that involves minimum walking from SAP Center to MLK Jr library if you've to depart in an hour."*

After performing this task, we requested users to fill out a survey to understand the quantitative feedback about the responses. Following images show the survey responses:

Post Survey Questions

| | Strongly Disagree | Disagree | Neutral | Agree | Strongly Agree | Standard Deviation | Responses | Weighted Average |
|---|---|---|---|---|---|---|---|---|
| The app was easy to navigate. | 0 (0%) | 0 (0%) | 0 (0%) | **4 (57%)** | 3 (43%) | 1.74 | 7 | 4.43 / 5 |
| The app solved my problem (getting route information from A to B) | 0 (0%) | 0 (0%) | 1 (14%) | **5 (71%)** | 1 (14%) | 1.85 | 7 | 4 / 5 |
| I would use this app again. | 0 (0%) | 0 (0%) | 1 (14%) | **5 (71%)** | 1 (14%) | 1.85 | 7 | 4 / 5 |
| I would recommend it to a friend. | 0 (0%) | 0 (0%) | 3 (43%) | **4 (57%)** | 0 (0%) | 1.74 | 7 | 3.57 / 5 |
| It was easy to navigate my way around the app. | 0 (0%) | 0 (0%) | 1 (14%) | **5 (71%)** | 1 (14%) | 1.85 | 7 | 4 / 5 |
| I could get the information quickly. | 0 (0%) | 0 (0%) | 2 (29%) | **4 (57%)** | 1 (14%) | 1.5 | 7 | 3.86 / 5 |
| It is easy to remember where to find things. | 0 (0%) | 1 (14%) | 0 (0%) | **4 (57%)** | 2 (29%) | 1.5 | 7 | 4 / 5 |
| Information is layered effectively across screens. | 0 (0%) | 0 (0%) | 1 (14%) | **6 (86%)** | 0 (0%) | 2.33 | 7 | 3.86 / 5 |
| Some screens were cluttered with information. | 0 (0%) | 2 (29%) | **4 (57%)** | 1 (14%) | 0 (0%) | 1.5 | 7 | 2.86 / 5 |
| My mistakes were easy to correct/ or revert/ or rollback. | 0 (0%) | 0 (0%) | 2 (29%) | **3 (43%)** | 2 (29%) | 1.2 | 7 | 4 / 5 |
| | | | | | | | | 3.86 / 5 |

Fig. 6. Survey responses

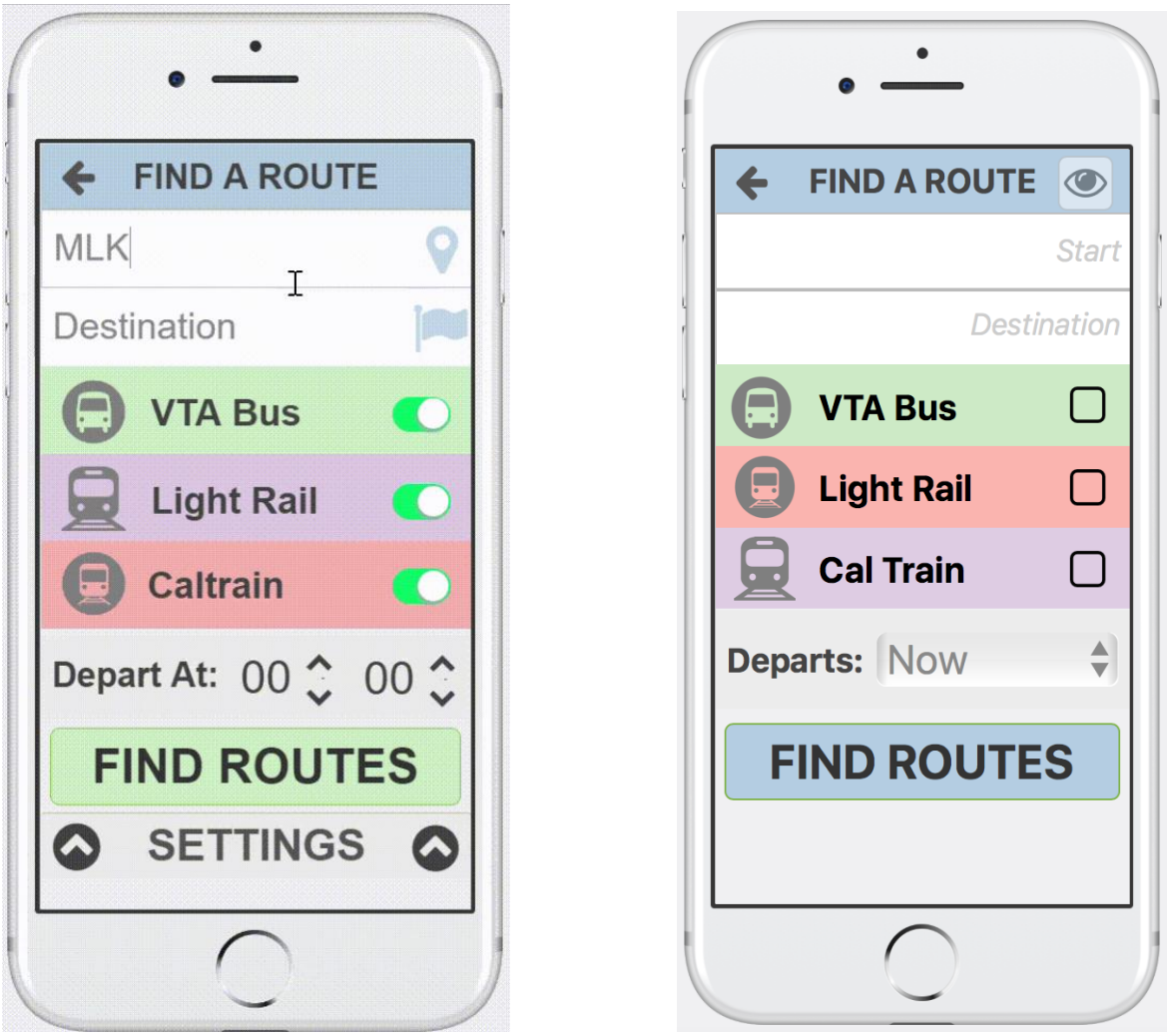
## Incorporating feedback

Whole purpose of conducting user testing was to get the user feedback. We had tested around a dozen users. We had two basic sources of collected feedback

1. Observation notes
2. Survey responses

We addressed almost all the concerns raised by users to make the application user experience as smooth as possible. Following were the major concerns raised:

Concern 1: From the survey responses, we had gathered that some screens were cluttered with the information
Our response: We removed the unnecessary information
Concern 2: Settings button was occupying too much space on the screen and it was not adding lot of value to the screen.
Our response: We removed the setting button and added new icon on the top of the screen
Concern 3: “Departs At” was a scroller with 24 values for hours and 60 values for minutes. Users didn’t find it very efficient to use scrollers.
Our response: Removed scrollers and added drop down which has values as “Now”, “In 1 hour” , “In 2 hours” and “In 4 hours”.




a) Previous b) Current

Fig.7. Refining the design by incorporating user feedback

## Future Scope

Future work involves having more than two color schemes. We are thinking of having at least a half dozen new color schemes specifically chosen for each type of color blindness. The user can try various options and chose the one which suits their preference.

## Promotion and Marketing

Because people with compromised vision are in the minority, our promotional activity and marketing has to be targeted advertising. As our targeted users are travellers with compromised vision, we are thinking of posting flyers at an ophthalmologist clinic. Posting advertisement on sites which have information related to accessibility will also be helpful. Along with this, our promotional strategy involves putting up flyers with catchy phrases at transit locations like bus stops, light rail, and train stations.

## Conclusion

This has been a learning experience to see an idea go through multiple iterations and evolve into a refined prototype. This project provided an opportunity to apply all the HCI principles that we learned in the course. This experience has empowered us to take on more challenging and complex projects and also bring to life ideas of our own.

## Building a successful product: The Journey!

Our project went through several iterations and evolutions before arriving at its final version. At the start of the prototyping process, we all had different ideas for how the product should look and work. Each team member approached the paper prototype independently, without consulting each other. This resulted in a variety of different designs, some of which worked better than others. However, the basic framework of the final app was based on Vikram's paper prototype with several new features suggested by the other team members, such as the addition of a map.

Once we had agreed on the basic design of the app, the next step was selecting a tool to create our hi-fi prototype. Most of us had experience using Atomic from the in-class Temperature Converter exercise, and Atomic was originally one of the finalists in our choice of which prototyping tool to use. Once we found that the other contender, Pixate, was no longer available we decided to focus on learning to use Atomic. However, even though Atomic had many useful features including variables and conditional statements, some of the group members felt that its learning curve was too steep. We decide to try another tool, Mockingbot, and see how it compared to Atomic. Mockingbot gave gratifying results more quickly than Atomic, and so this is the tool we settled on to use for our hi-fi prototype.

We divided work on the prototype evenly between the team, with each team member working on 1-2 separate pages of the mock-up. This gave the app an inconsistent look however, and caused issues during the first user test. Our initial prototype suffered from problems with the consistency and accuracy of data within the app, as well as general interface problems that confused some users. By the second user test, we had addressed some of the problems we found in the first, including the inconsistent data. We also improved the consistency between screens.

During the period after user testing we continued to make small improvements to the app along with one radical change. At the suggestion of an HFE student, we completely dropped our original idea of having three separate color palettes for each of the three types of colorblindness, and instead opted for the simpler solution of allowing the user to switch to a single high-contrast mode guaranteed to be visible to all: black and white. We quickly created black and white versions for each page of the app. Though this was a last minute change decided about a week before the final version was due, it has become one of the central features of our app.

After having completely different ideas about how our app should work at the start of the design process, we eventually came together around a single vision. Through a long series of revisions, removals and a good deal of trial and error, our app slowly evolved over time into its final form.