PROTOTYPE "THIS = THEN = THAT"

Martin-John Hearty - GitHub: https://github.com/MartinJohnH/Cart360
Instructor(s): Elio Bidinost & Sabine Rosenberg
CART 360 - Section AA

Link to Progressive Web App (PWA): https://tracket-a1e05.web.app/

The prototyping phase is crucial as it determines what works and what does not. It informs you on where you need to go and what needs more work. Iteration is key as you need to build your project gradually and not be afraid to change it while developing it. This can be difficult as I sometimes found myself to be a bit stubborn when it came to make changes to my initial plan. My prototype had to change based on what I thought it was going to look like in my mind.

My prototype can be subdivided into two sections, a physical one and a digital one. For my electronic and physical prototype, I was able to develop it at low fidelity level. I designed the necessary circuits, tested some sensors and electronics as well as tested some code. I had some difficulties with the heart rate sensor has it is not very stable and the data is difficult to interpret. I would like to use a wireless heart rate monitor which seems more promising. Subsequently, my digital part of my project is at a medium to high fidelity level of development.

I also have to work on the connection between the particle and the PWA, to do so I will connect it via a SQLite database. I'm very pleased with the level of progress for my web app. I put a good amount of time designing and creating layouts for it. I also took some time to think about the best ways to visualize the data I'm getting from my jacket. I found that a horizontal bar graph is the best solution. I did quite a few iterations on my designs for my PWA. I worked on layouts, colours, gestures, etc. More about the design and progress of my prototype can be found on the following pages.

PROTOTYPE "THIS = THEN = THAT"

For my project, I will be using two sensors, a heart rate sensor and an Ultimate GPS Module. Combined, they will send data to a Progressive Web App (PWA). The heartbeat module will be responsible for collecting data about the user's heart rate. Once this data is processed by the microcontroller, it will be sent to a database. I'm following this tutorial on how to obtain relevant data from the heartbeat module to determine one's heart beat: https://create.arduino.cc/projecthub/Johan_Ha/from-ky-039-to-heart-rate-0abfca. For my final project, I would like to use a wireless heart rate monitor which looks like it produces better results. The Ultimate GPS Module will be collecting the user's location. With the information obtained by the GPS, I can use the locations and time to calculate the speed and distance of the user as speed is a function of distance over time. There is one button that activates the jacket placed near the chest area. It will be used to turn the garment on and off.

Once the jacket is connected to the internet, it will send the collected data to a SQLite database using a PHP script. To access the database and connect it to the front-end of my web app, I will use GraphQL to query the data, display it using React components and then package the hole as a Progressive Web App using Gatsby.js. This will insure that my PWA is fast, optimized for the web and be functional offline. Additionally, I will be using Sass to develop the front-end part of my app.

I want the physical interaction with my jacket to be very minimal. The jacket itself will respond to the input of the sensors by turning on and off some RGB LED. However, there will only be a few as I want it to be subtle. I do not want to look like a cyborg or a light-up as a Christmas tree. I want to be able to wear my jacket, turn it one and forget about the rest.

The main interaction with the project is with a smartphone. The collected data is visualized with bar graphs as I found them to be the easiest way for a user to see their progress.

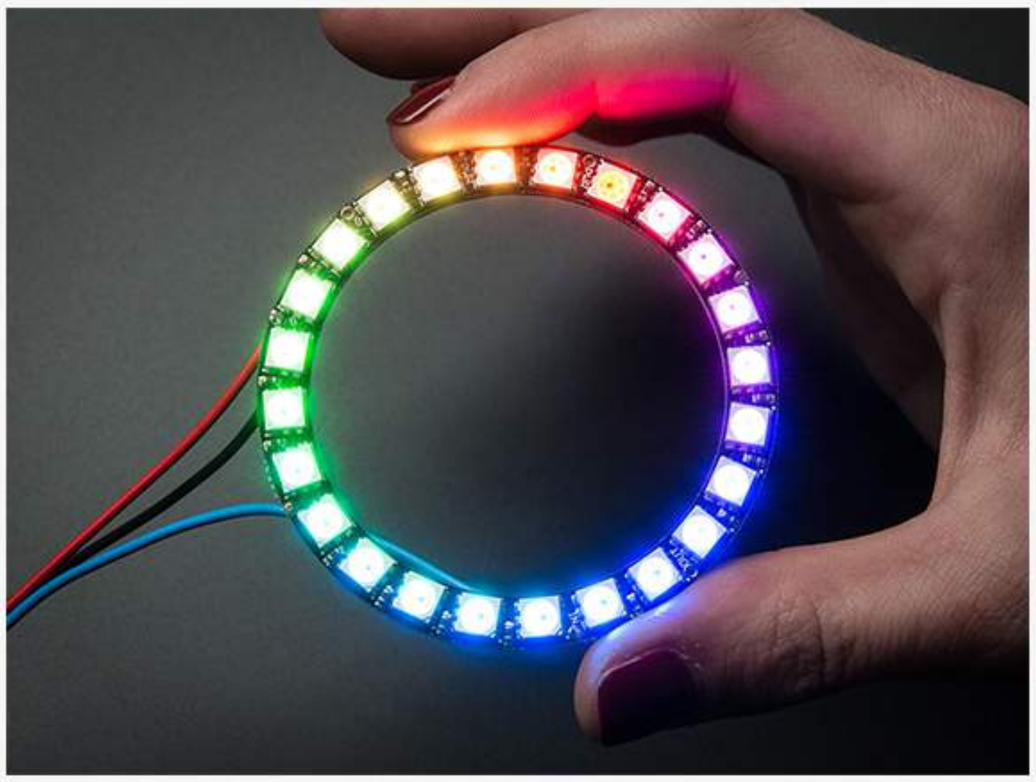
PROTOTYPE "THIS = THEN = THAT"

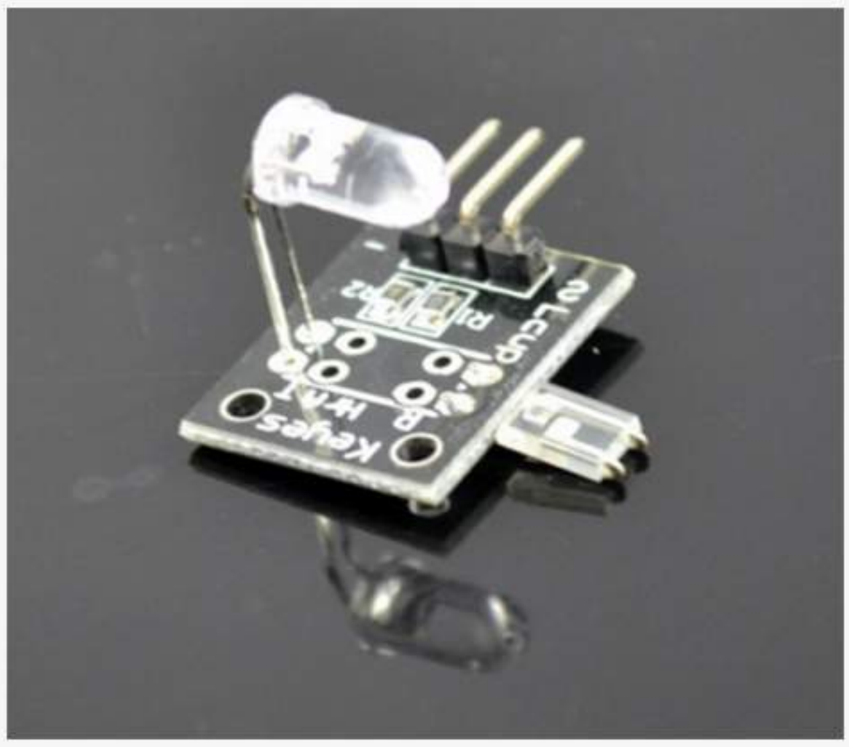
My prototype has indeed evolved and changed over the past few weeks. I decided to remove some other parts of the jacket, including the morse code. I found it to be more of a gimmick and did not serve any purpose. I also decided to create a Progressive Web App to go along with the project.

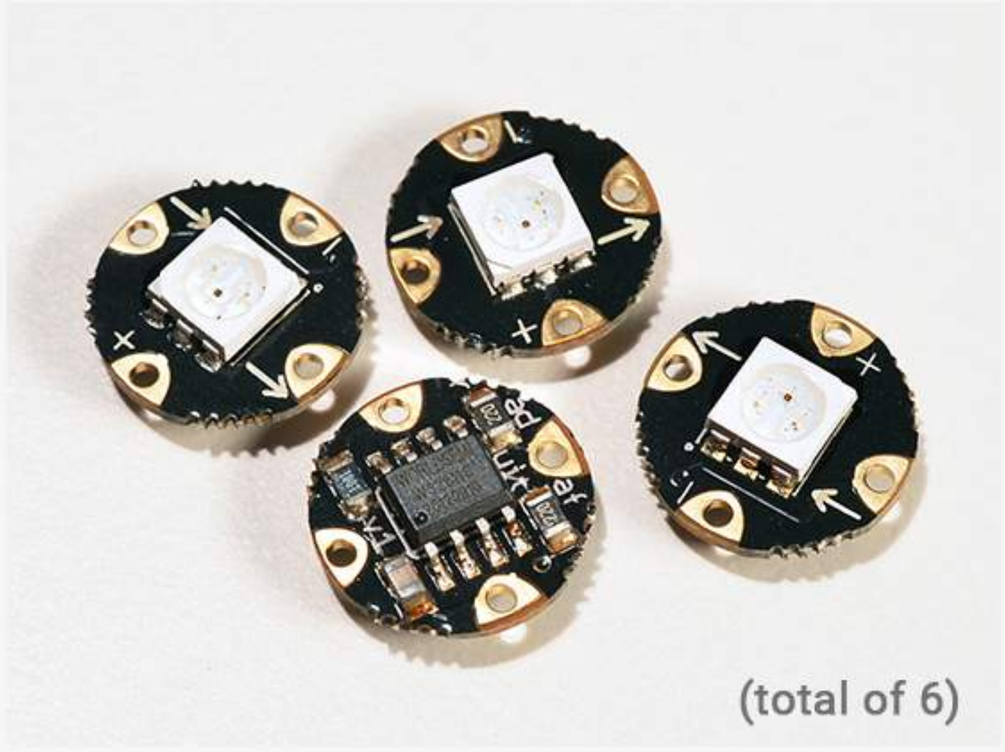
When thinking about the PWA, I thought that it needed more attention then the jacket as it will be the main entry of interaction. My jacket has become a sensorial garment that sends data to a web app. The PWA will display three types of data: distance, speed and pulse. Each data type will then be connected to another web page that will display a bar graph representing the user's weekly progress. When the user pinches the screen, they will be presented with their monthly progress. The app will also indicate when the jacket is connected.

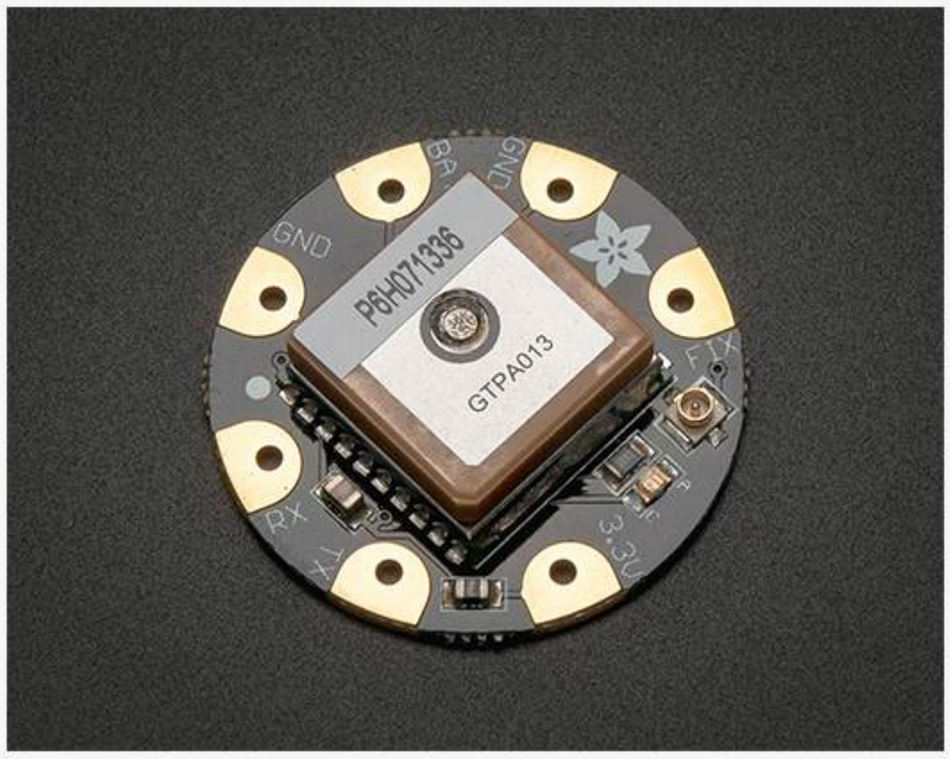
As for the physical part of my project, apart from removing some elements, not much as changed. The heart sensor and GPS are still main sensors. There will still be a neopixel ring, but I have removed a few RGB LEDs around the pocket area. There will be a total of six LEDs on the jacket instead of nine.

Electronics

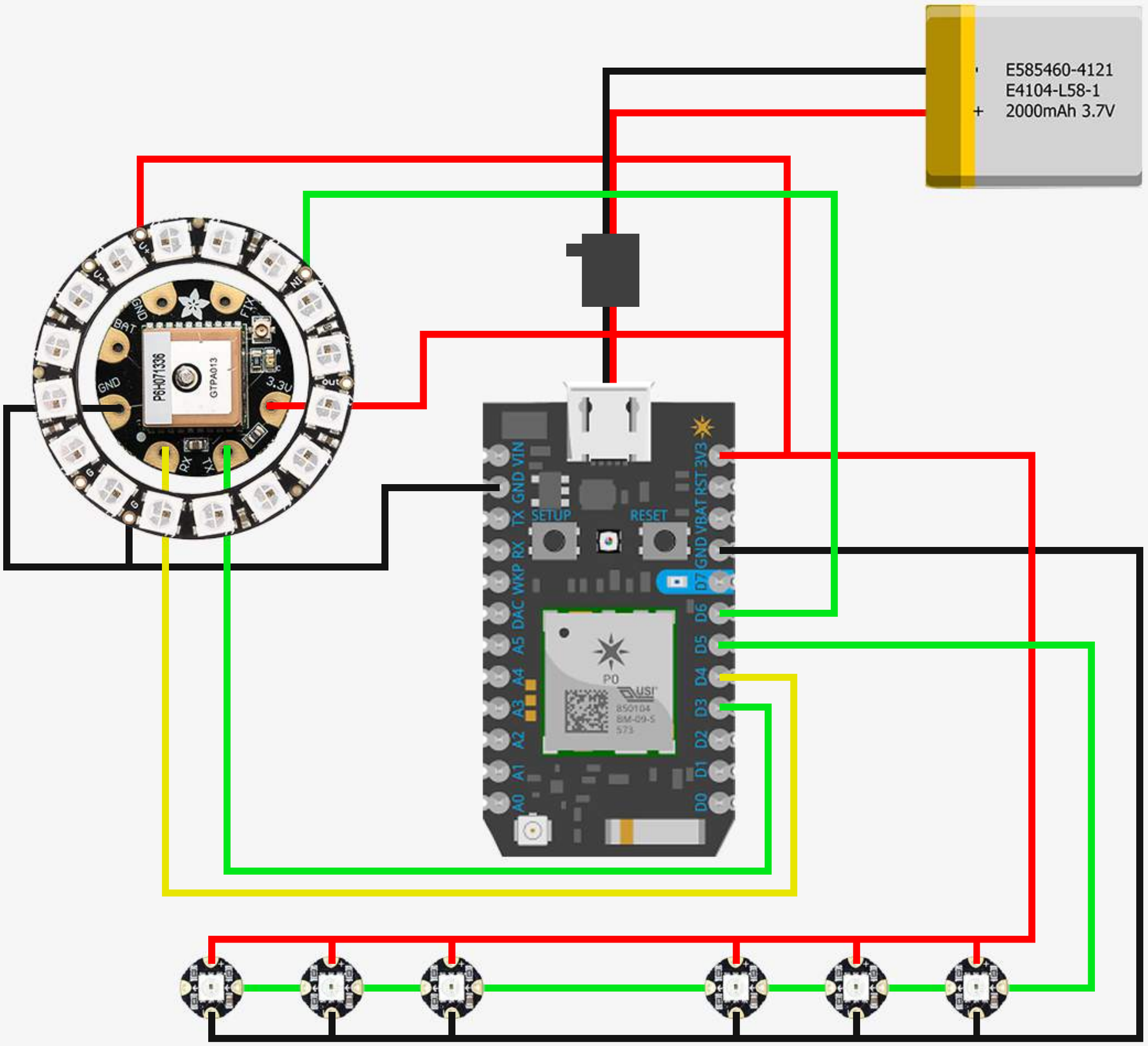




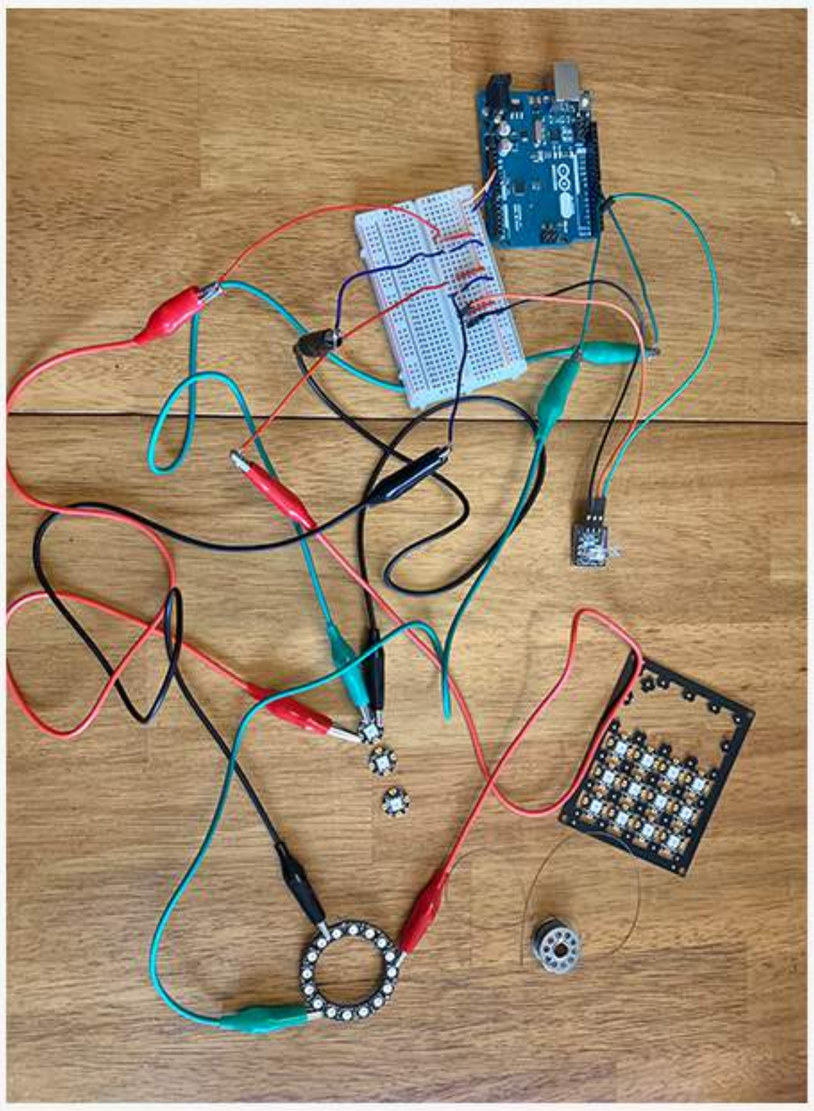


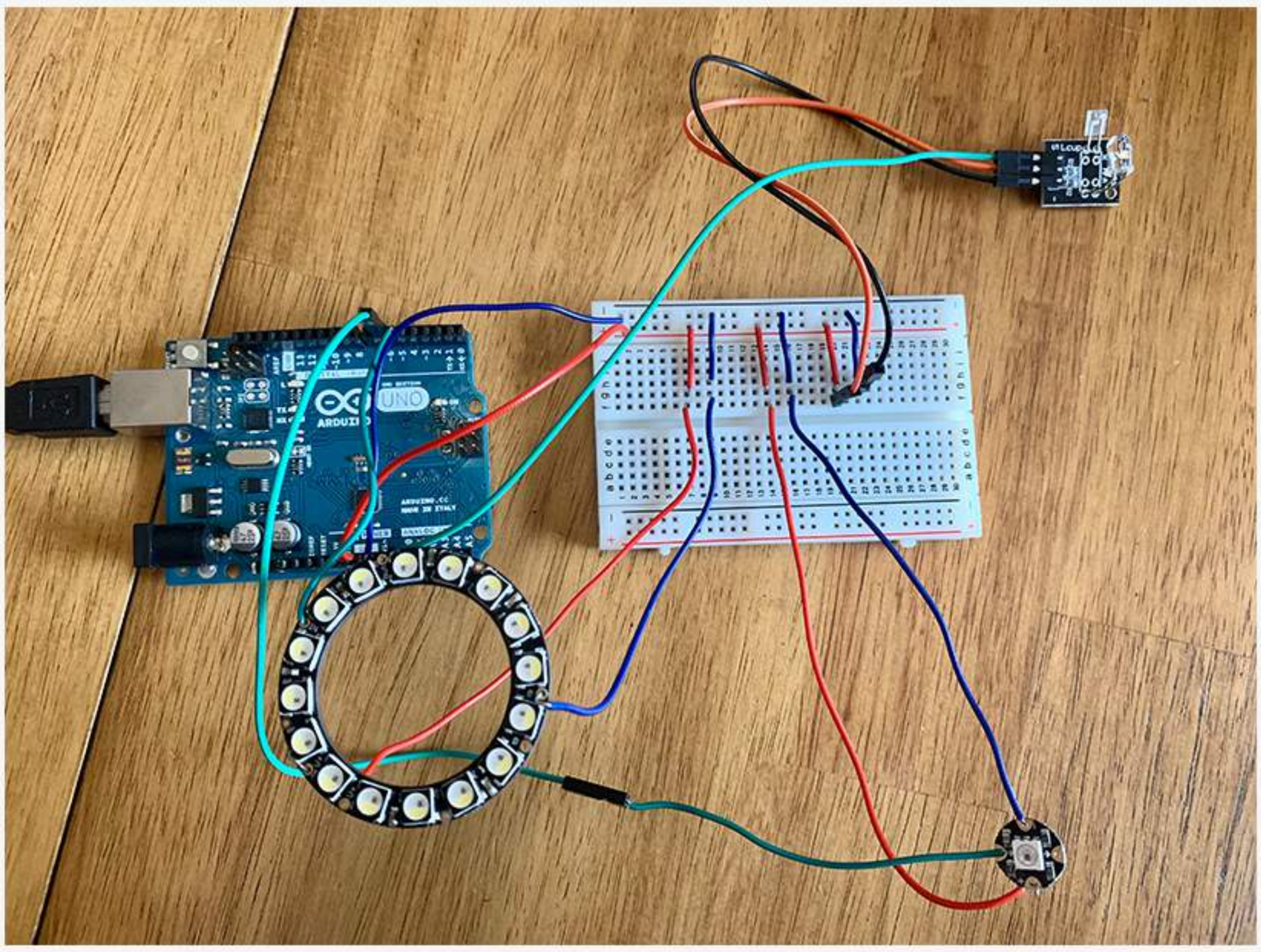


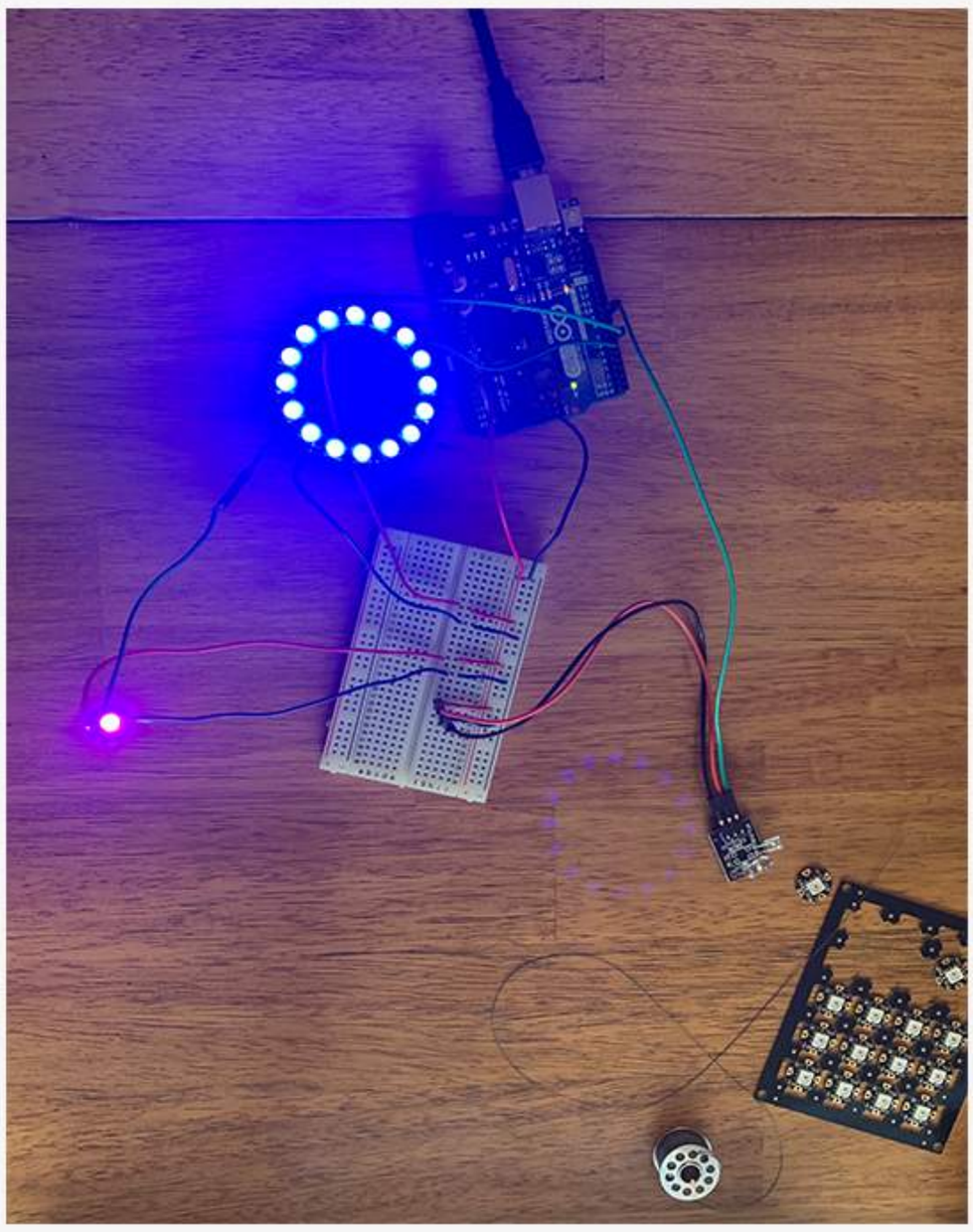
Electronic circuit

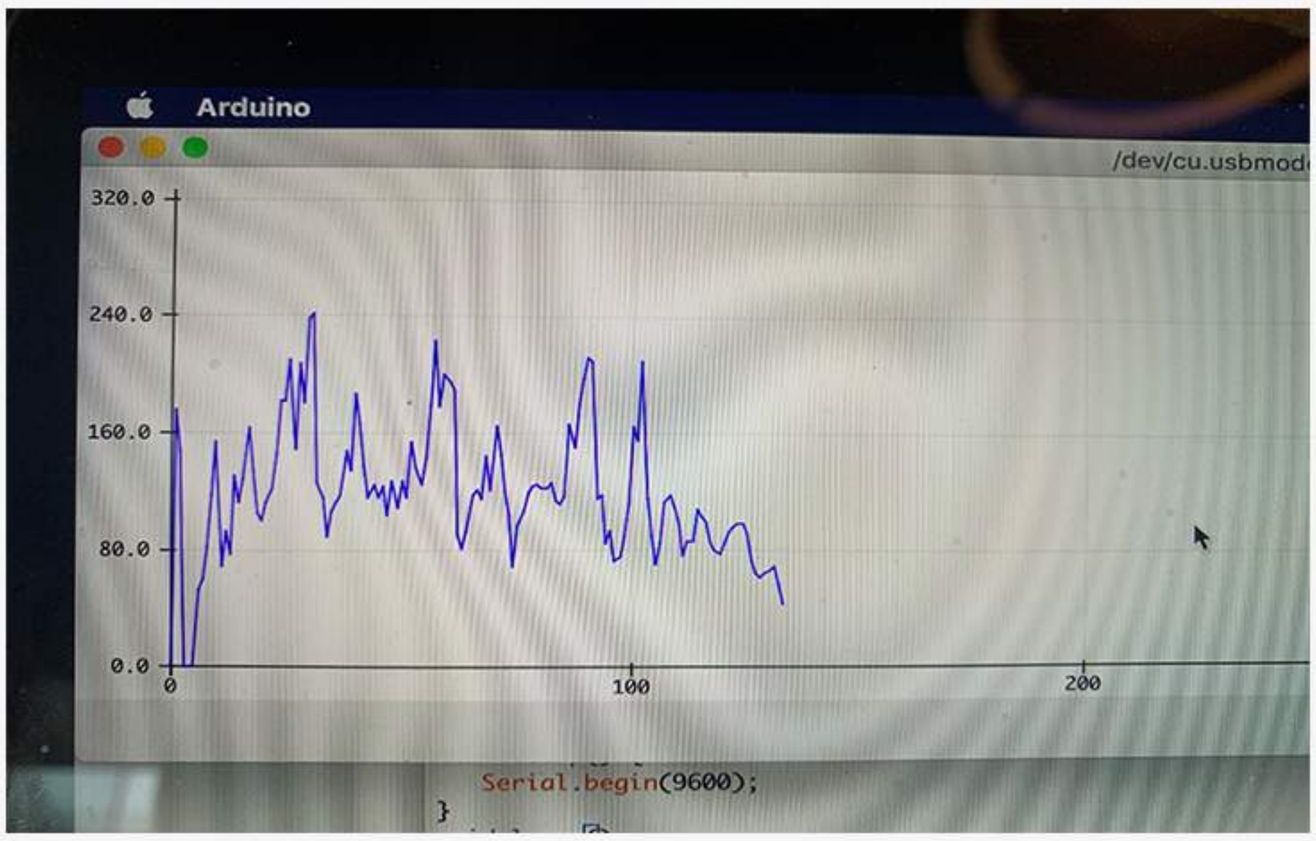


Progress



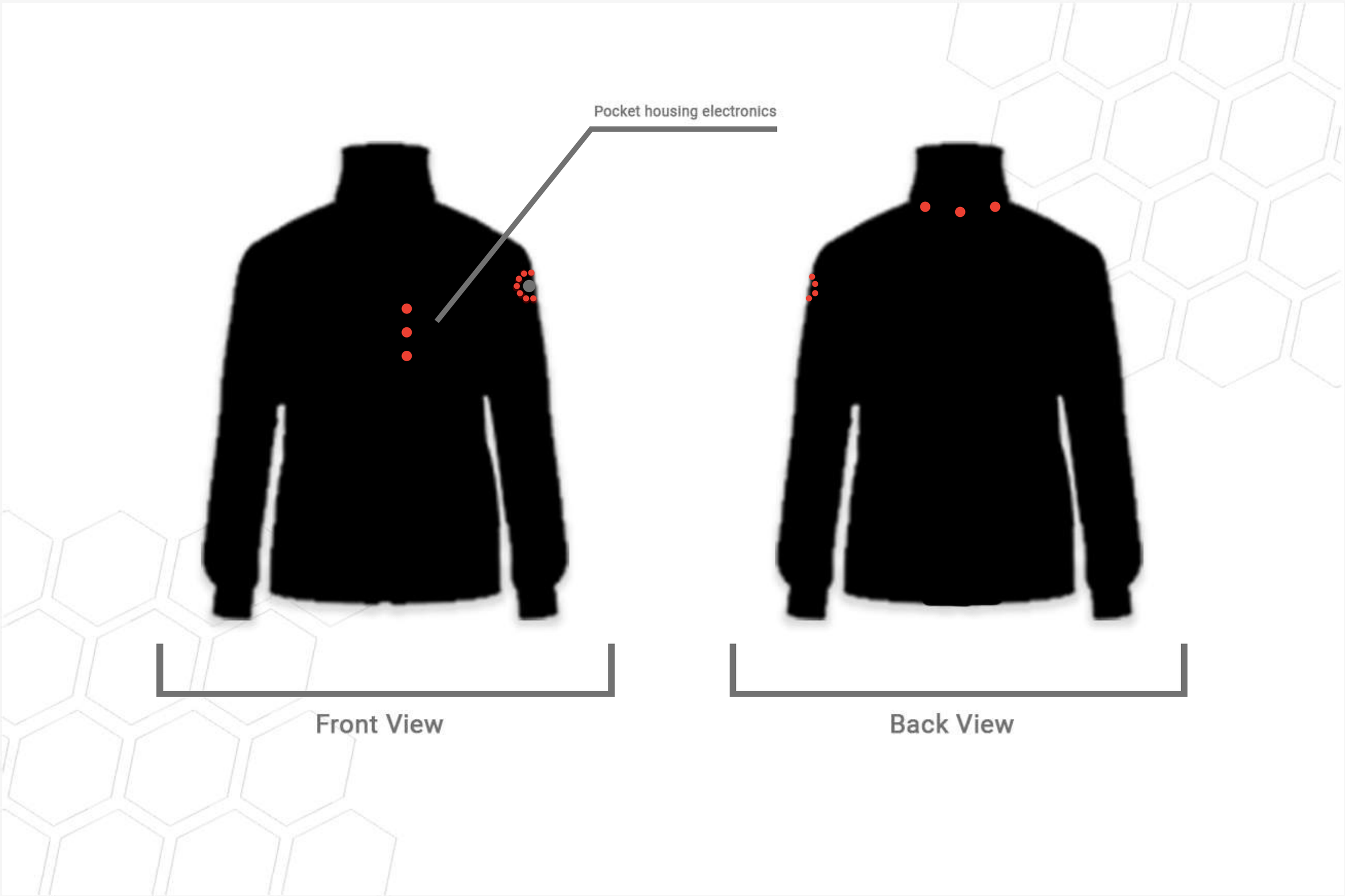






Data from heart rate sensor

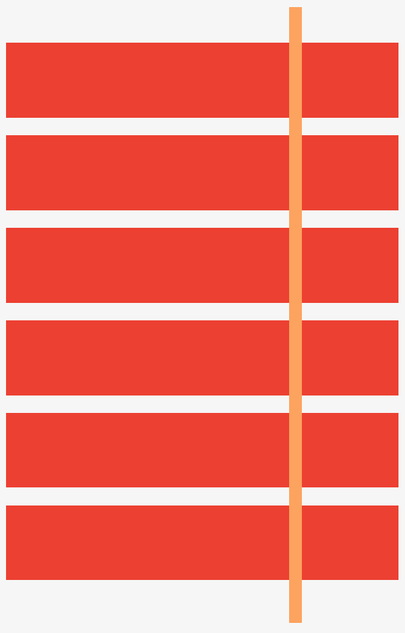
Jacket

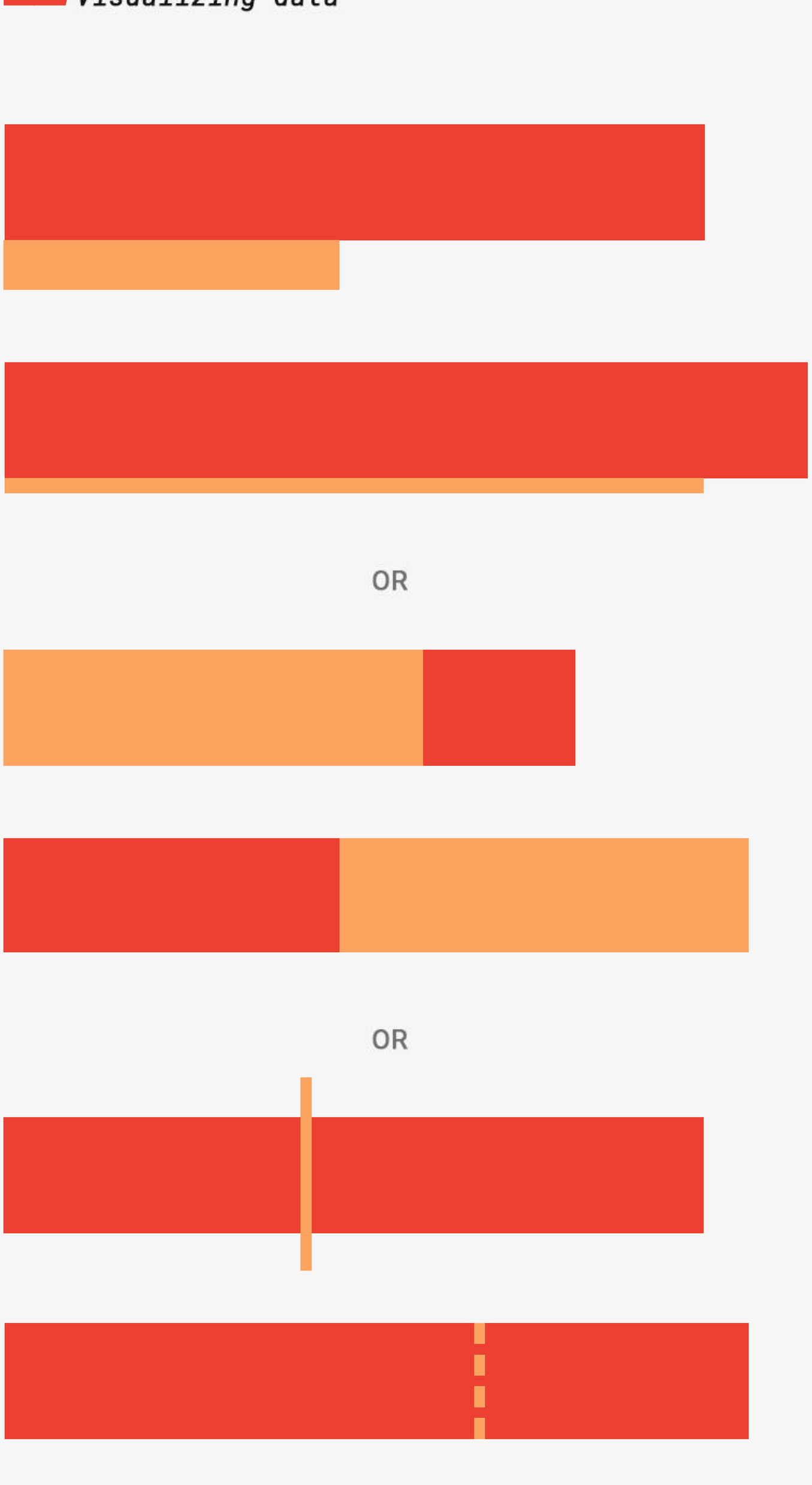

Pocket housing electronics
Front View
Back View


POLAR
T34
POLAR
M

Colours

HEX:#0C0C0C HEX:#141414 HEX:#EC4032 HEX:#EC6358 HEX:#FAA45F



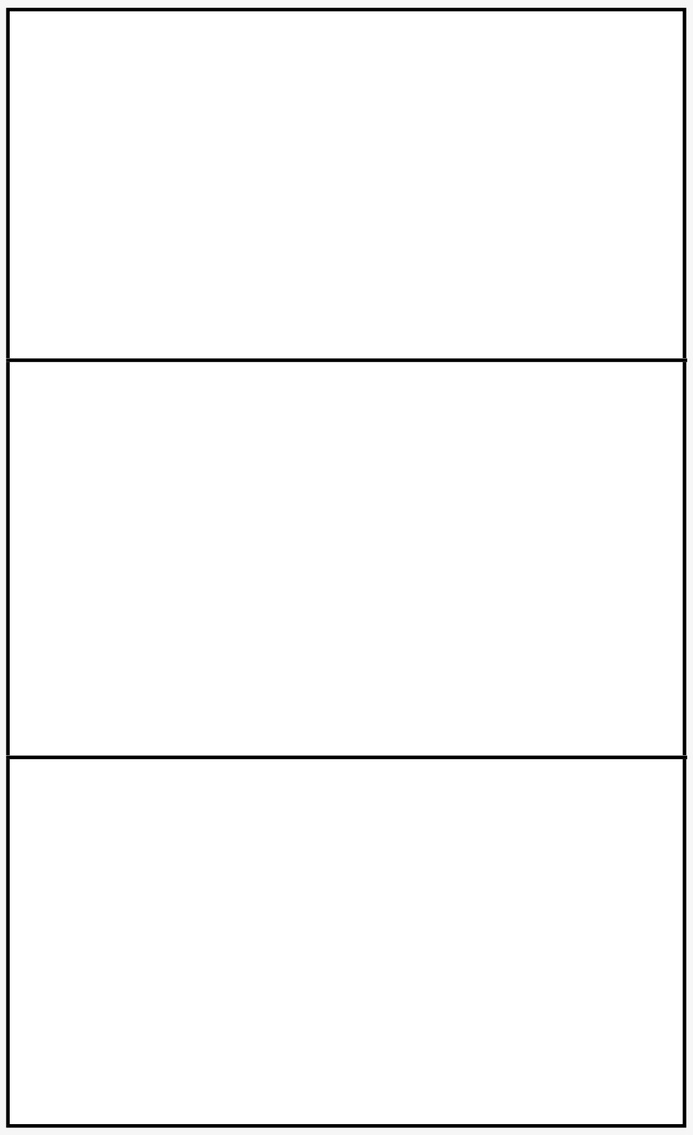


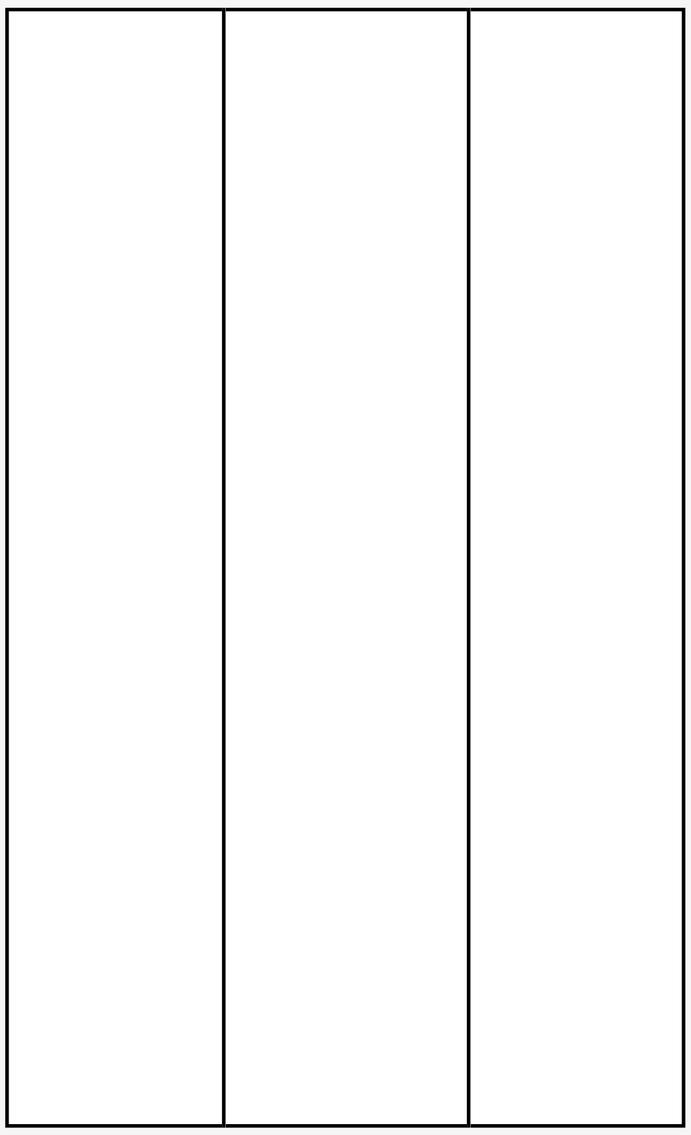
Takes up a bit of space
Easy to compare

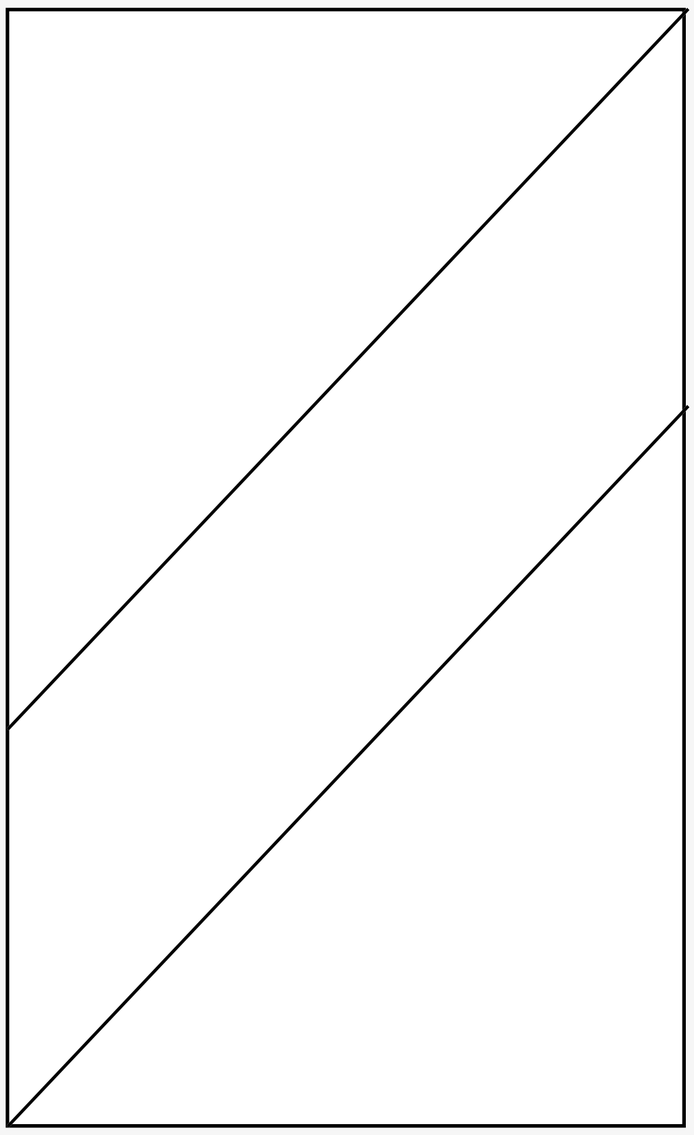
Takes up less space
Not as easy to comapre

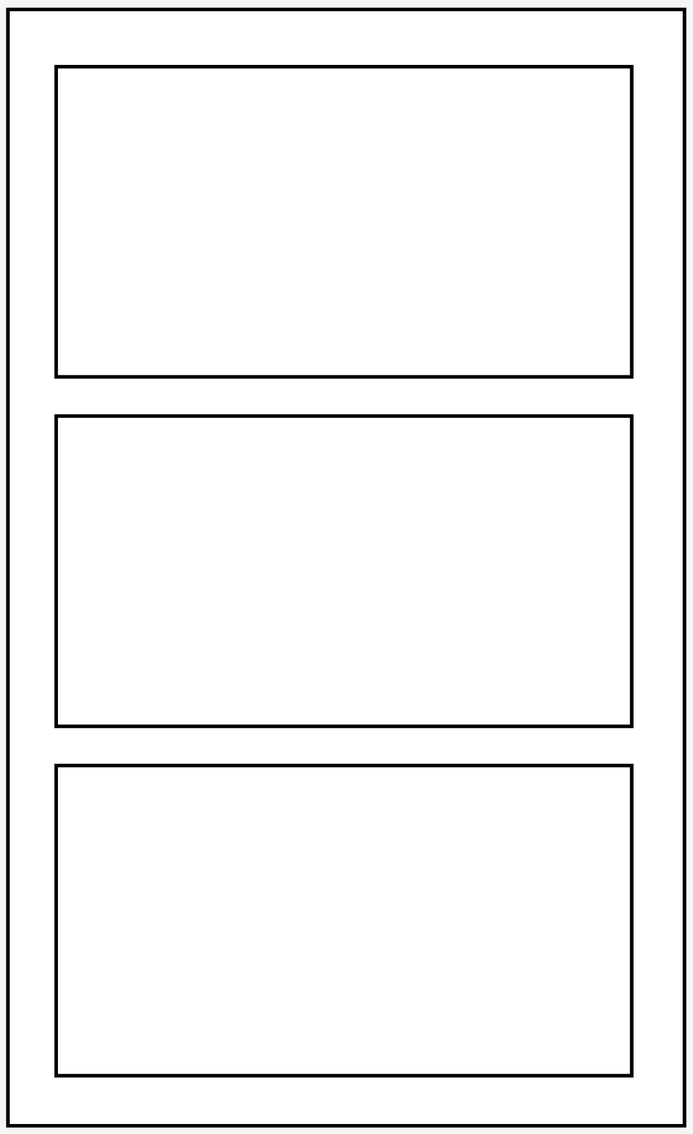
Takes up less space
Soso to compare

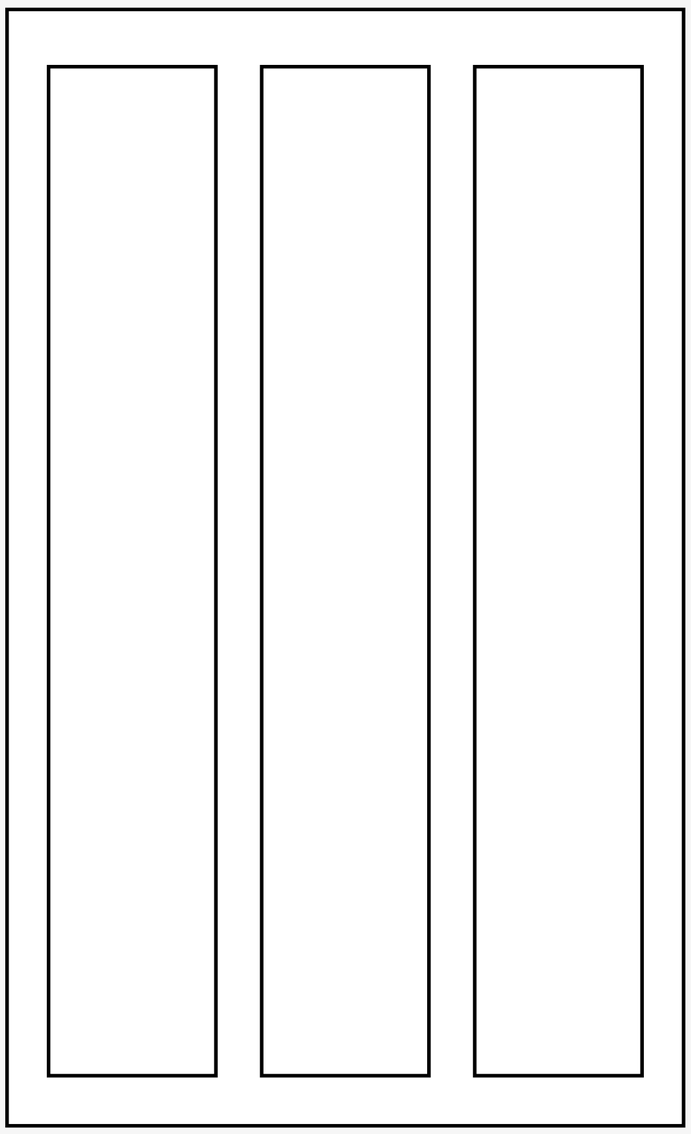
Possible layouts

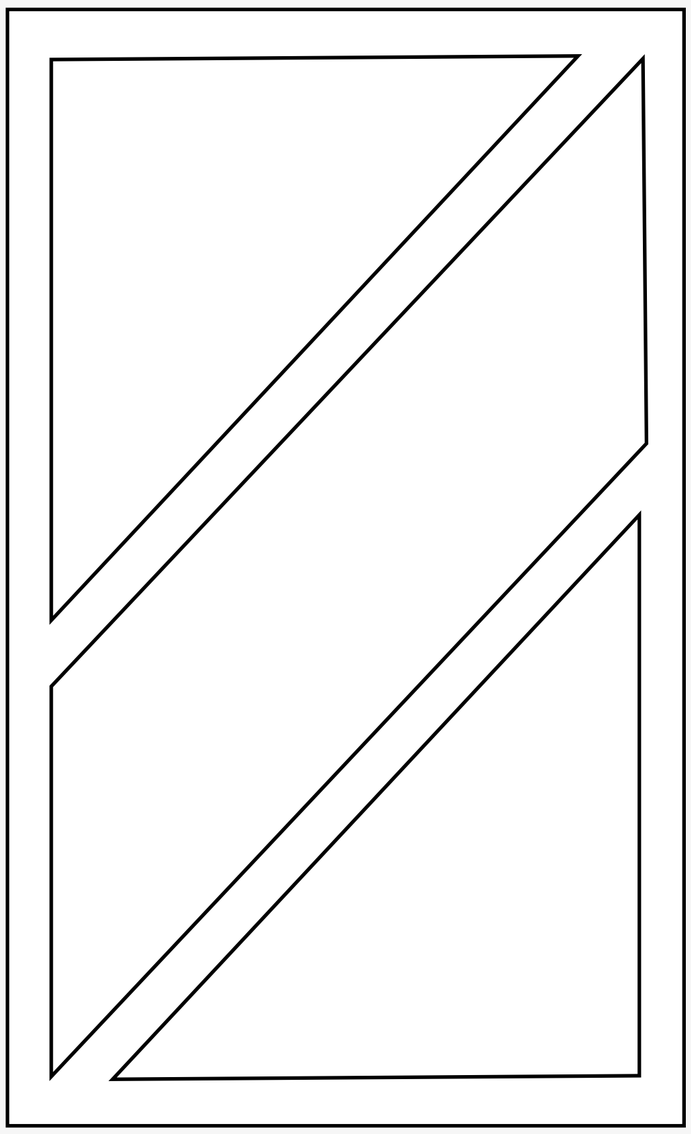




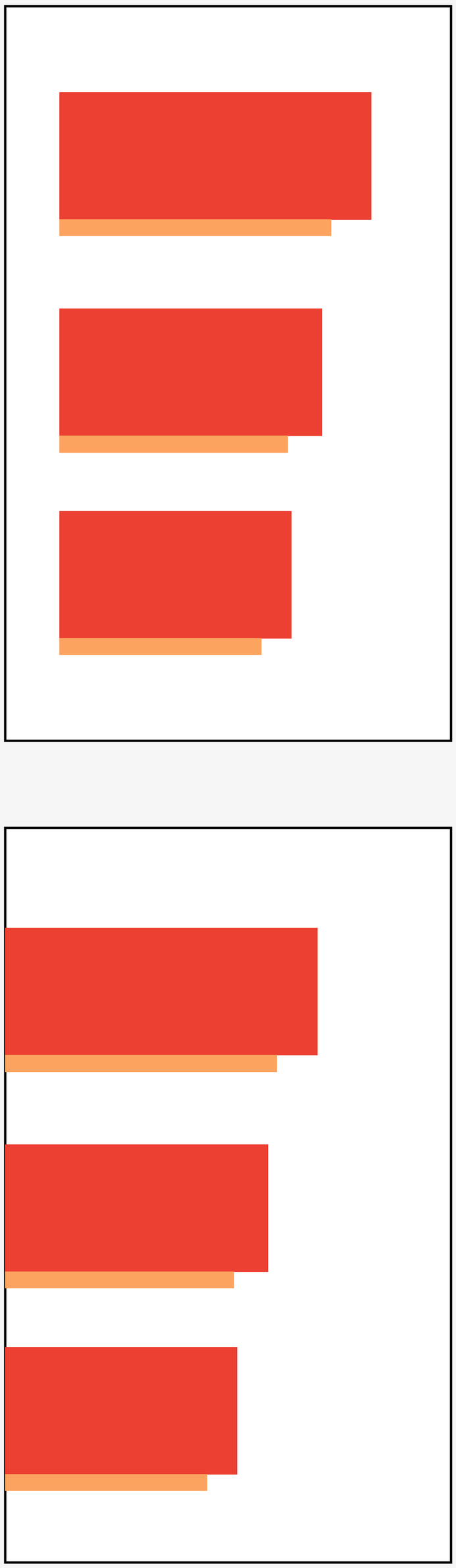


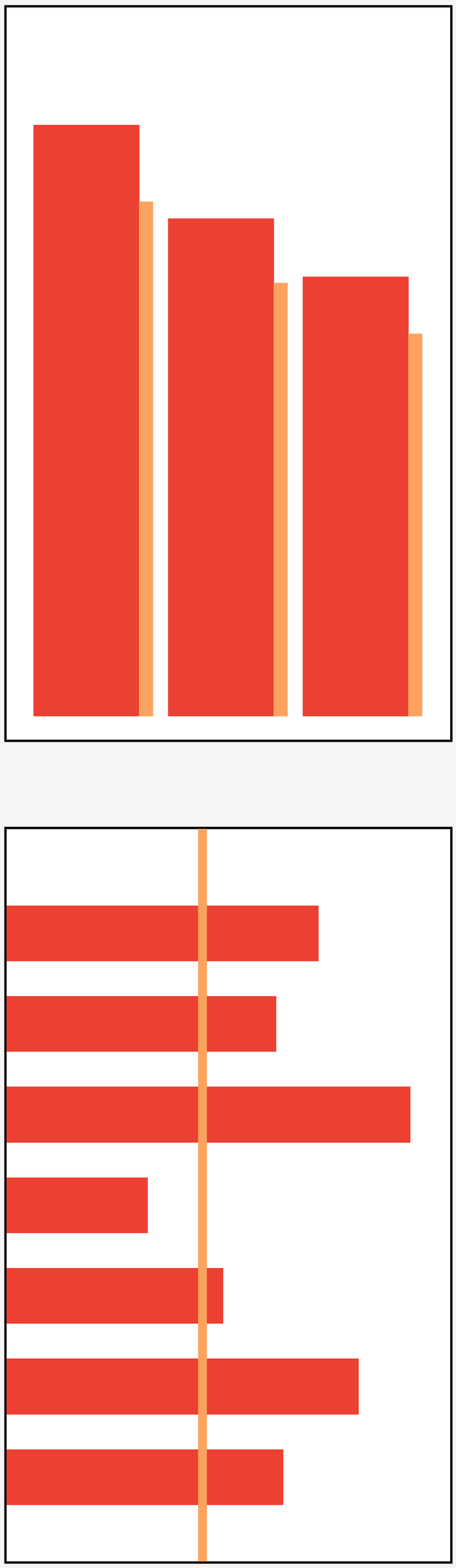


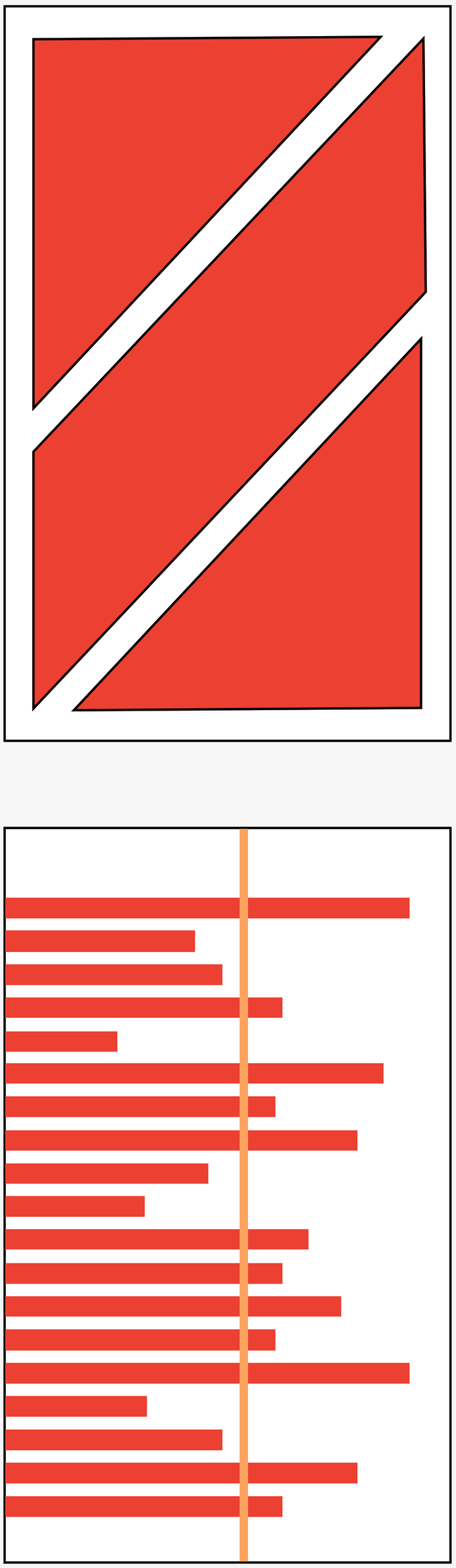




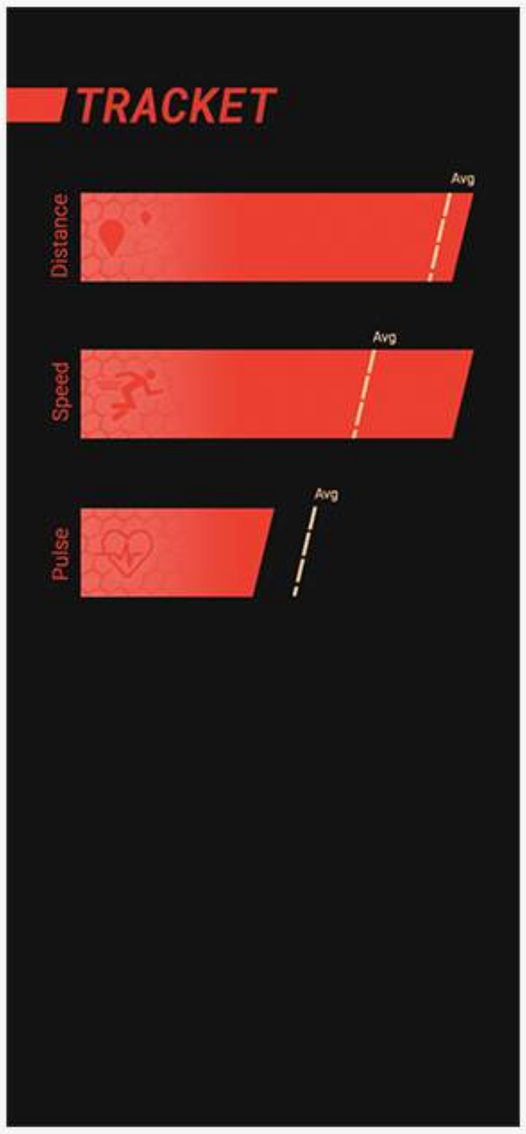
Possible layouts cont'd

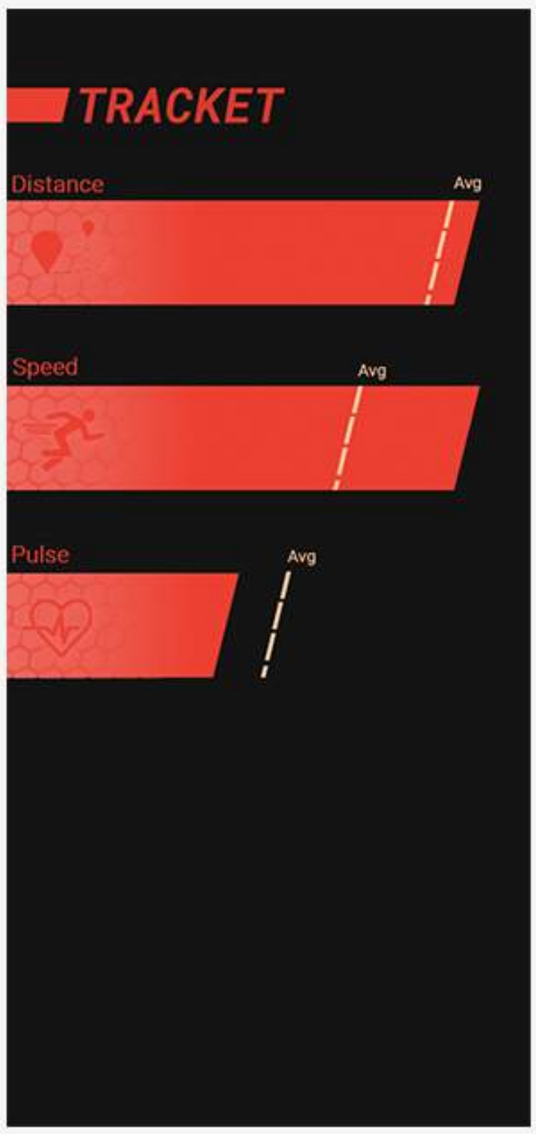


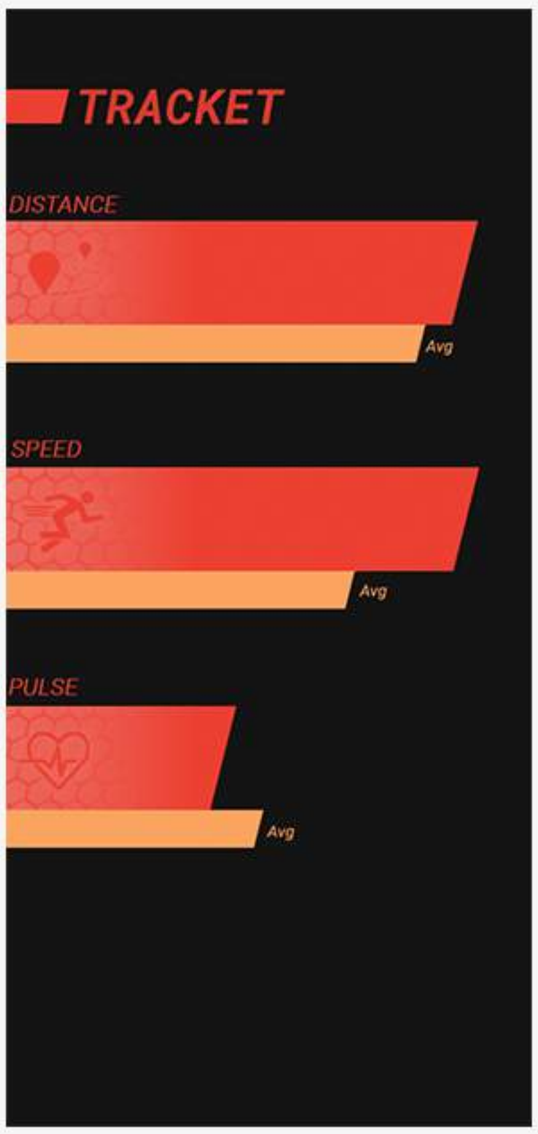




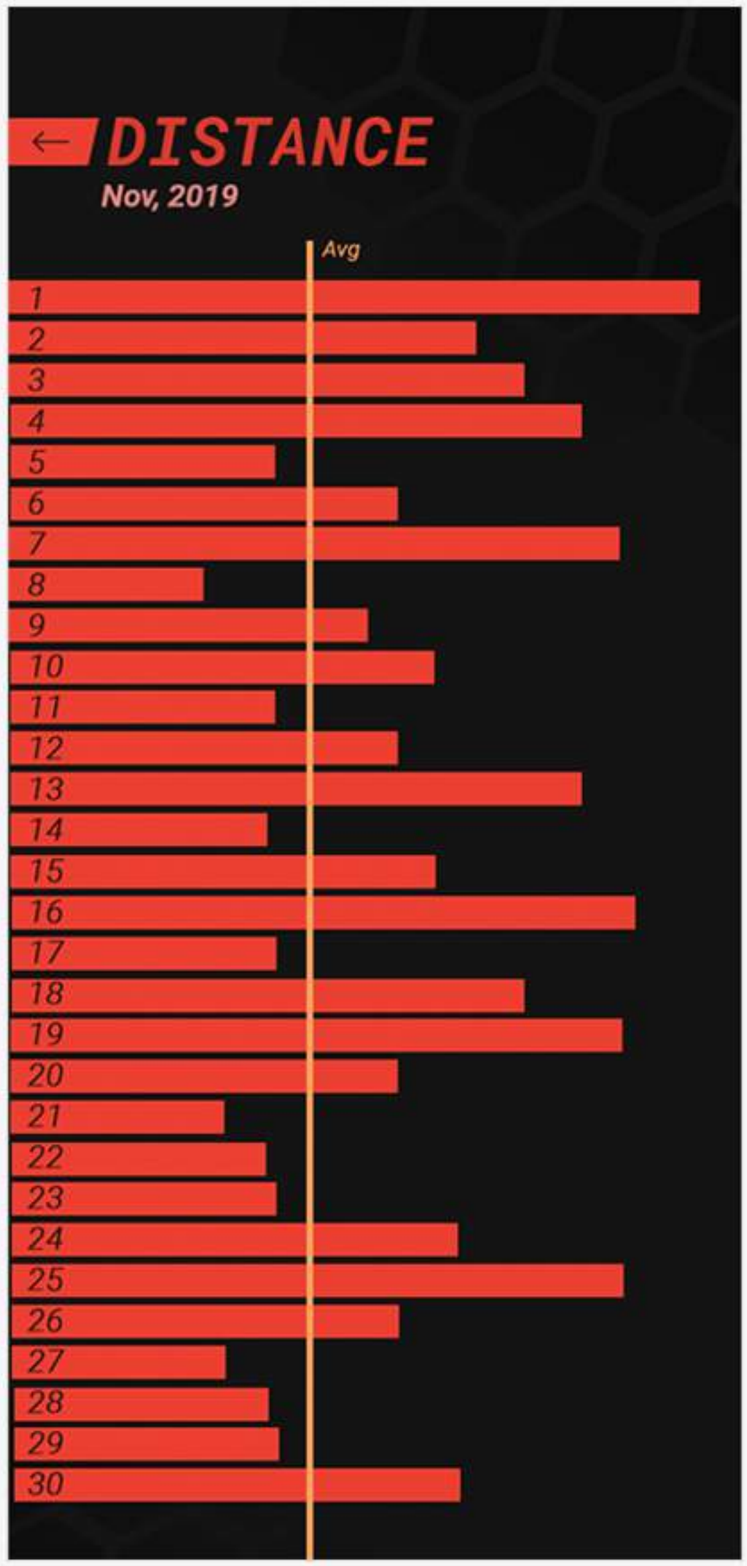
Design evolution

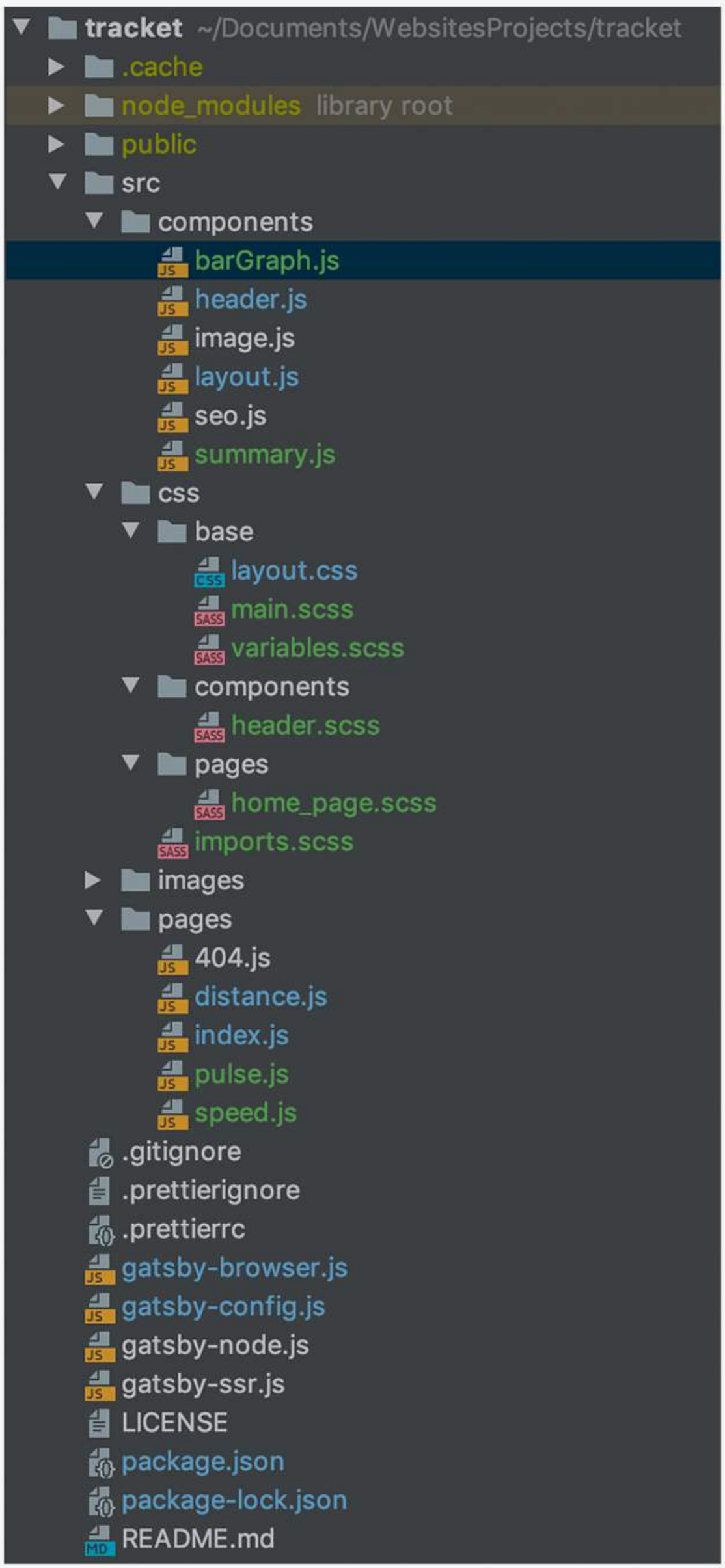






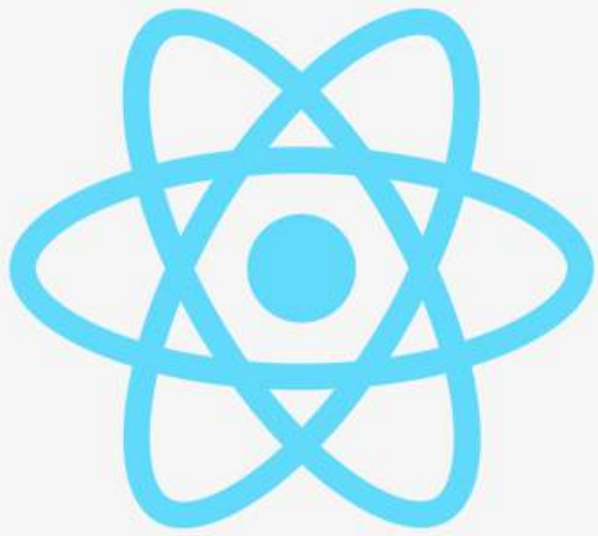
Final design

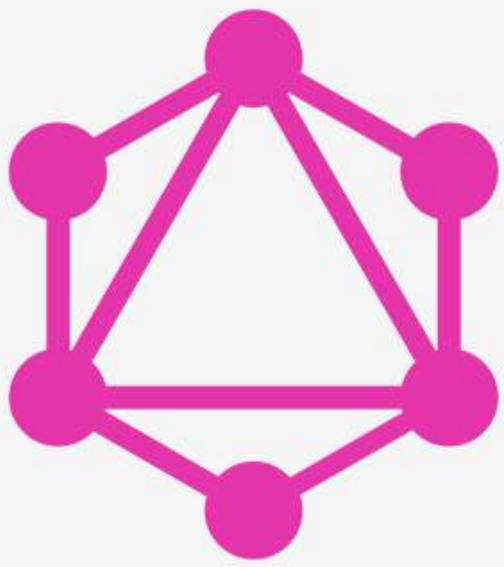




GatsbyJS

ReactJS





GraphQL

Sass