the process behind creating awear:
a mobile app for real-time physio monitoring

by Ahmad Soomro

*when I speak in the first person, please note none of this would be possible without the guidance of Dr. Adam Strang, my super-chill mentor who came up with the idea for the application.

background

Most Air Force operations take place in teams. It's important for commanders to have situational awareness about airmen health stats to increase efficiency and reduce risk. There was a need for a mobile application that monitors team physiology in real-time so that commanders could know how specific airmen's physiology was responding in training sessions and operations. This application should also serve as a field-grade research tool (my mentor particularly wanted to use it to study collaboration and coordination).

The US Air Force Research Laboratory's 711th Human Performance Wing wanted to develop this app. So, they brought me on as a Repperger Scholar to work on it!

designing a seamless user experience

1 identifying core functionalities

I first identified exactly what I want this app to do. I wanted it to:

a) pick up beats per minute (BPM) sensor data from heart rate monitors ,
b) visualize this data,
c) record this data so it can be analyzed later

I built the whole app by expanding on these three core functionalities.

Then the question was about *usability*. How could get the app to satisfy these three core functionalities, while making it easy to use?

According to the Healthcare Information and Management Systems Society's (HIMSS) work group on usability[1]:

- The data should be the focus.
- The layout should contain typographic elements (i.e. icons) that communicate meaning and are consistent across screens.
- Feedback should be easy to record.
- Interaction should deal with the task at hand.

2 setting up a basic app layout using wireframes

Using HIMSS' requirements for usability, I landed on a four page layout.

<u>These are the page titles: *Live*, *Sensors*, *Records*, and *Setup*.</u>

- If you want to see the status of sensor data and record a session, you go to the *Live* (main) page (the data should be the focus).
- If you want to add/remove a sensor, you go to the *Sensors* screen.
- If you want details about a previous session recording, you go to the *Records* screen.
- If you want to add an airman and edit their info (such as their weight, age, etc.), you use the *Setup* page. (We used weight info to calculate body mass index to determine heart rate ranges)

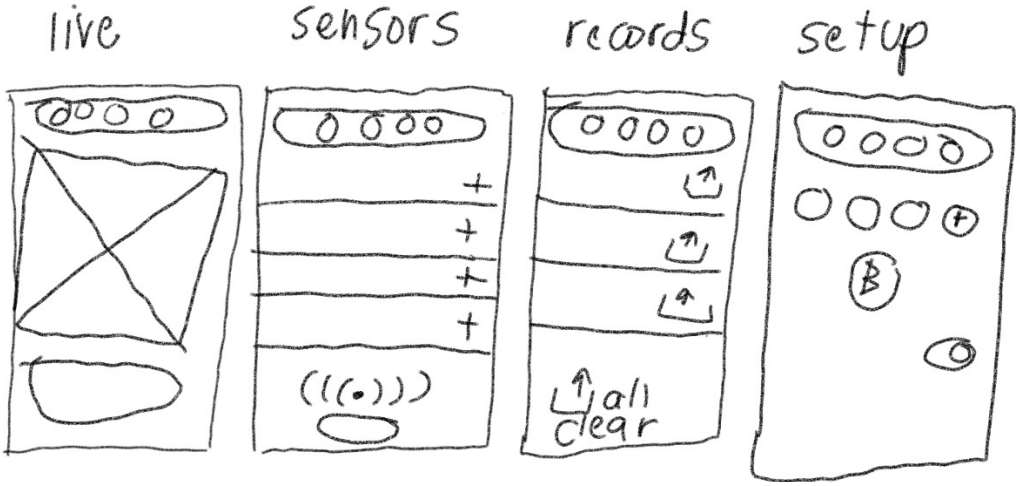

Figure 1: A very simple wireframe of what pages I needed (these are not the actual wireframes I used, just an example)

I laid out a really simple wireframe so I would know exactly what pages I needed and could visualize them. This figure does not include wireframes for the popups (to edit airmen info, to throw errors, etc.).

3 picking colors

Aside from HIMSS' standard, another part of the app being usable involves color choice[2]. I wanted a color scheme that was familiar, yet easy on the eye. I wanted the data visualization colors to be intuitive as well so that just by looking at the color of a number, a commander can know whether the airman's BPM was in normal range (I'm going to discuss my thoughts around viewing the data more later).

Initially, my mentor wanted me to use the Air Force's official colors:

...but I just ignored his request. They were too dark and grungy. I wanted something lighter, inviting, and versatile.

When I was searching for a color palette I had a few things in mind: The palette needed to flow well with the red-orange-yellow-green (hot to cold) color schema that I wanted to use for the heart rate data visualization (these colors are the most important because they are the colors that are part of the data visualization!).

***It goes without even having to explain that RED means the heart rate is too high, and the GREEN would indicate a more relaxed heart rate. I later added blue to indicate a heart rate at rest.*

I also wanted to make it easy to recognize airmen's data without reading their name, so I wanted the commander to assign a color badge to each airman. I knew my base colors would have to work well with these color badges as well.

The colors shown below were the bases, but eventually I made them more subdued so they didn't pop out over the colors of the numbers that were meant to indicate heart rate.

The colors I ended up settling with for the base colors for the whole app were:

These colors flowed well with the heart rate visualization colors, and reminded me of air. Get it? *Air* Force. I eventually used a gradient of these colors for the background because it just looked so airy, cloudy, and easy-on-the-eye.

4 picking icons and other symbols

One of the HIMSS usability criteria is making sure the layout uses typographic elements that communicate meaning and are consistent across screens.

I wanted to pick icons for the nvaigation bar that were intuitive so that anyone could navigate the app without being told how to.

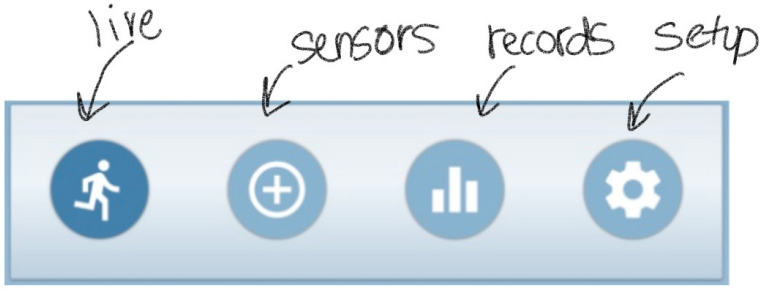

Figure 2 A super-intuitive navigation bar

What makes a navigation bar easy to use is the familiarity of the icons and an indicator of the current screen.

Something else that was important to me was using an animation to symbolize that the app was searching for sensors. When the user clicks *search* on the sensors page, I didn't want to leave them guessing as to whether the app is searching or not. So I used a simple pulse animation to indicate the app was indeed searching for sensors.

5 visualizing the data

The most important screen, in my opinion, is the *Live* screen. When an activity is underway, this is the screen being viewed. Airmen wear heart rate monitoring sensors and their heart rate in BPM (or percentage) is displayed. The numbers are big, easy to read, and color coded.

Number colors correspond to heart rate zones. These zones help reach fitness and athletic goals more efficiently3. For example, zone one (green) serves as a warm-up zone while zone five (red) serves as a high-intensity zone. If you're resting your information is just displayed in blue. This data can also be viewed as a percentage of the maximum heart rate, as shown in Figure 2.

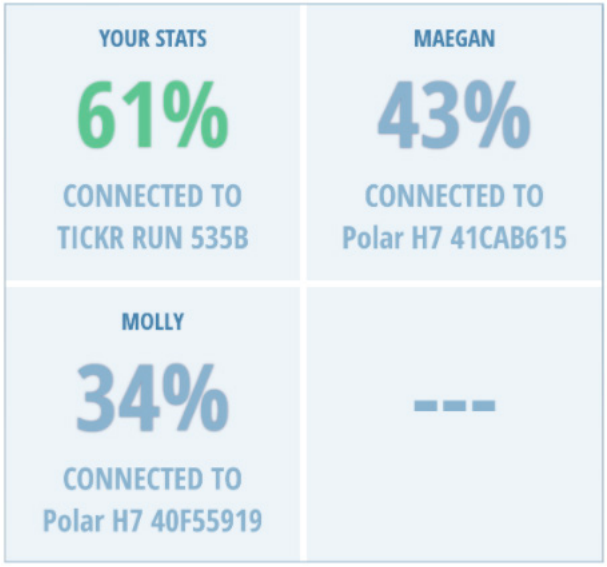

Figure 3 A lively *Live* screen

Not only does the user see the data, but they can easily see who it belongs to and which sensor it's coming from.

I realized that reading the names and sensors was not really a good way to visualize the data, so I created a feature that allowed the user to assign color badges to each airmen, so they didn't have tor read the name to know who the number belonged to.

I also wanted to create a + icon on the *Live* screen on any empty boxes so that the user could add an airman into that box without having to navigate to the *Setup* page.

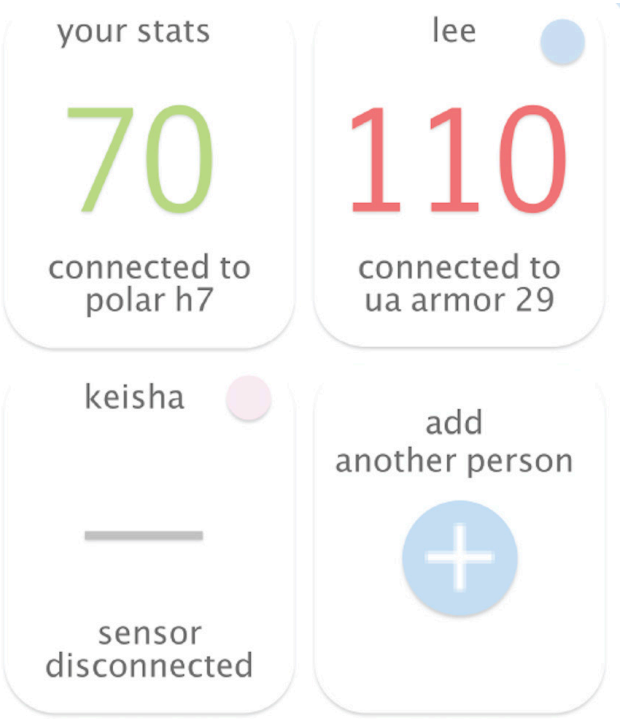

Figure 4 The *Live* screen, updated

This newer screen now shows color badges and a + icon that lets the user add an airman without having to navigate to the *Setup* page. Also, notice the design has changed a bit, I'll go over this a bit later.

The reason there were only four slots was because when writing the code, I was only able to handle four sensors at a time. If the app could handle more sensors, I would lay this data out as a list (after there were more than four users) and have the list automatically show airmen's whose heart rate was the highest at the top.

6 integrating the app with bluetooth sensors

For the majority of the project, I was working with Underarmour's (UA) sensor API, and it was a nightmare. Our lab was actually partnered with UA, so my mentor was sort of keen on trying to brand the app as integrated with UA tech. But I just wasn't having it because I had to make a UA account, use an OAUTH token, be connected to the Internet to validate that token...

I basically had to ignore my mentor's preference again! I just found a way to pair the bluetooth signal from ANY heart rate sensor. This way, you don't need an Internet connection to make the app work, and as long as the sensor transmitted a Bluetooth signal, my app could pick it up.

My mentor was really happy with this because we had so many different brands of sensors in the lab, and they all ended up pairing with my app without a fluke.

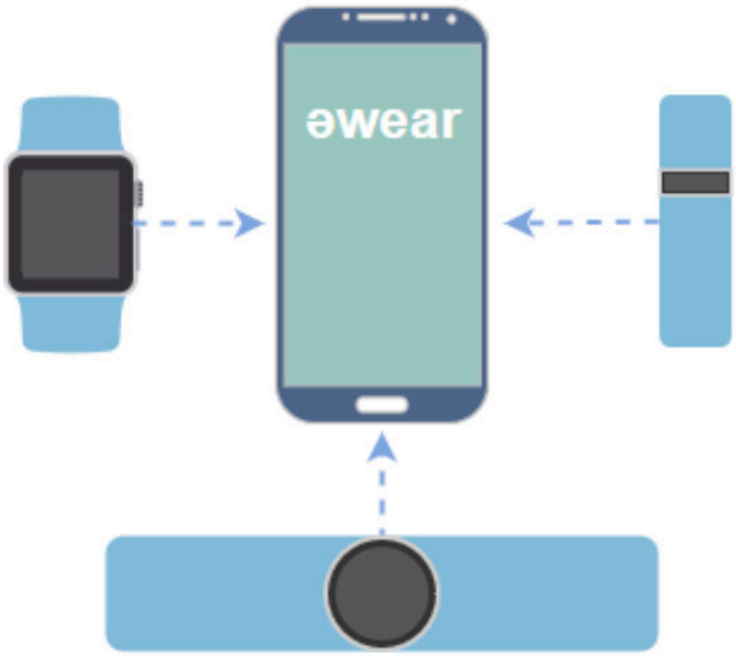

Figure 5 If it's BLE, aware can read it

aware can picks up fitbits, polars, UA's, and just about any brand of sensor that can transmit a Bluetooth LE signal

7 recording session data, notes, and feedback

Every session that was recorded has a corresponding .CSV file. These files are saved on the device and can be exported through micro USB or the email connected to the device (if it has Internet access).

Feedback and session notes are able to be captured through the *Live* screen using a simple text box element. The text that's in the box is saved at the end of the session.

Each connected user's sensor data is recorded into the .CSV file every 10 seconds and timestamped, along with the session title, and the notes.

Users can navigate to the *Records* page to edit session names, delete the session, or share.

8 offering customizability options

On the *Setup* page, the user can decide whether they want the heart rates on the *Live* page to be displayed in BPMs or percentages (relative to the body mass index of each connected airman).

The app also allows each connected user to be connected to a color of choice (from the color options shown in the portion of this article where I talk about colors).

conclusion

In this section of the article I want to sort of talk about the limitations of this app, how it could be improved for the future, and share some of the mockups (I'm unable to share actual pictures of the app because I wasn't given permission to share the actual app).

Limitations:

Time: I only had about 2.5 months to finish the project. On top of that, I didn't even know how to program in Android. So during this 2 months, I had to learn Android Programming while designing. It was a lot to handle, but at the end of my stay, I was the only scholar who presented a complete, working demo. I was also the only scholar invited back.

Sensor capaity: Ideally, I would prefer that the app can handle more sensors—at least 12. But I was able to get 4 sensors, and for my mentor that was good enough.

Visualizing records: I really wanted to have a feature that allowed me to visualize the records within the app using line graphs, but due to time constraints I was not able to work on this part of the project.

Adapting the transfer protocol for combat scenarios: In a combat scenario, if Bluetooth signals were going off, this can be dangerous because it could give up soldier locations. Ideally, the app would use heart rate monitors that transmit a signal thats not able to be recognized by devices that are not authenticated.

Accurate geolocation: What would be cool is if the *Live* screen was a map that used geolocation of each device. I'm there's a way to do this, but because that wasn't one of the core functionalities, I didn't make it a priority.

Overall, the app was a success. Everyone loved it, many of the other researchers at the 711th Human Performance Wing wanted me to work on their projects next, and I was very humbled.

Things I didn't discuss to keep things concise: how I chose the name, why I made the logo the way I made it, the different kinds of buttons on the app, and why I made the setup page the way I made it. If you have any questions about these specific aspects, or have any other questions, feel free to shoot me an email at: designboyxyz@gmail.com!

Below I will share the old mockups of the app and the footnotes!! If you want to see the more recent mockups, they're available for view at: designboyxyz.github.io/aware. Thanks so much for taking the time to read this.

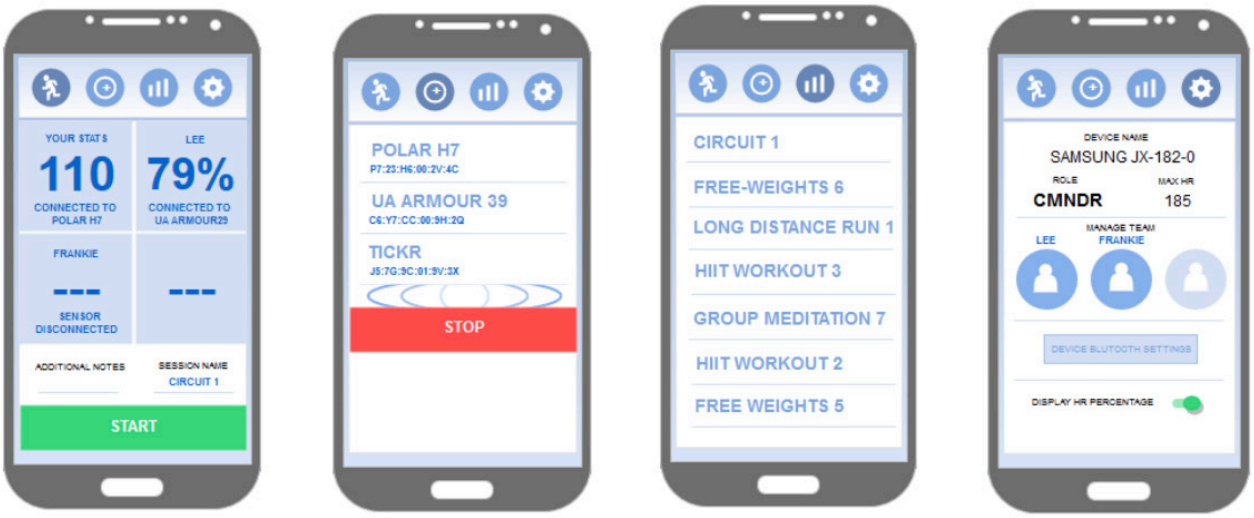

Figure 6 The old mockup
This is how aware started!!

---

1. HIMSS App Usability Work Group. "Selecting a Mobile App: Evaluating the Usability of Medical Applications." http://www.himss.org/selecting-mobile-app-evaluating-usability-medical-applications-0?ItemNumber=28900, pp. 4

2. Stickel, C. Maier, K. Ebner, M. Holzinger, A. "The Modeling of Harmonious Color Combinations for improved Usability and UX." Conference Proceedings of 31st International Conference Technology Interfaces (ITI 2009), D. 28, 2009, pp. 323-328

3. Cooper, E. "Optimizing Fitness Using Heart Rate Training Zones." Seattle Performance Medicine. http://www.precor.com/en-us/keep-me-moving/articles/aerobic-capacity-and-power-responses-cross-trai