Project Deliverable H: Prototype III and Customer Feedback

Team #7
GNG 1103C
Due Date: March 29, 2020
Abdel Baig (300121134)
Bilal Ahmad (300125852)
Ali Al-Zaidi (300082526)
David Onoruvwe (300144303)
Yazan Elmasry (300136709)

1. Introduction:

This deliverable will document the development and assessment of team 7's Prototype III test plan and includes the results obtained from the testing of the prototype as well as customer feedback. It will also include documentation of what was achieved during the term and what the team would have liked to achieve under better circumstances. Through the progression of prototype III, our team is responsible for testing the various components purchased, mentioned in project "Deliverable E". More specifically, we will be prototyping and testing the OP-Watch application by utilizing the most important subsystem, the measurement of heart rate and blood oxygen saturation levels by the pulse oximeter part, MAX30100, and the values sent to the application through bluetooth. Testing this subsystem will allow us to merge the GPS tracking system and the heart rate/oxygen saturation data with the calling feature in the future to alert emergency aid. This will ensure that our team has completed a sizable portion of the project, while also being aware of the risk and uncertainties associated with amalgamating some of the components together.

2. Prototyping Test Plan:

In prototype II, we created the circuit diagram of the heart and SpO2 subsystem on the Arduino breadboard. Our team tested this specific subsystem because we felt that this was the important one since the device's main functionality is to record the user's heart rate and oxygen saturation levels. The main purpose of this was to send the heart rate and SpO2 values from the breadboard to the Arduino's serial monitor screen. This was done successfully because when the user placed their finger on the sensor, the user's values were immediately sent to the serial monitor, thereby displaying the accuracy of our device.

This section will describe the team's developed test plan with regards to the third prototype.

2.1. Test Objectives Description:

For prototype III, our team has decided to create and test our application, entitled, "OP-Watch Companion App." The main function of this app is to call a user's loved one in the event that he/she has undergone an opioid overdose. This is arguably the most important feature of this device since the only way for the device to function is to call a dear loved one through the app.

The main objectives of this test are to analyse the application and to garner feedback from our ideas. The third prototype is composed of a physical and analytical model. The physical model is the Arduino circuit which holds all of the components together on the breadboard, and the analytical model is the OP-Watch app. For the physical model, we will utilise the Arduino software to send the heart rate and SpO2 numerical values to the application. The circuit diagram remains unchanged however, we will create a new set of code within the Arduino program that will allow us to send the values from the program to the desired app. Then, we will implement a calling and sound feature; two very important components of the app. Another key aspect of the device is that it will have a built-in GPS tracker. This will display the user's latitude and longitude values, which will aid in the emergency response team tracking their exact location, so they can successfully come to the victim's aid in the event of an overdose. The calling feature is also very important and it will be determined that when the user's heart rate and SpO2 levels are low, then the app will make a phone call to a loved one, who will then alert emergency services if need be. Whenever the phone call is being made, there will be a loud alarm sound that plays, thereby alerting others that an individual has overdosed. Once all of these objectives are met, the prototype is complete.

The possible types of results are on a yes/no basis. The first result is that when the SpO2 level is low (approx less than 95%), then the OP-Watch app makes a phone call to a user's loved one, thus sending the user's exact location to that specific individual. A loud emergency alarm sound will also play whenever the phone call is being made. This would clearly display that our setup is successful and that the prototype is considered complete. The second result is that whenever the SpO2 level is too low, the phone call is not made to a user's loved one. This indicates that the test is a failure, since the app is not working as we want it to. In the event that this occurs, we will ask for assistance from the TA's and if that is of little to no avail, we will debug the code used to create the application as well as the code in the Arduino program to pinpoint where the exact problem may be.

2.2. What is going on and how is it being done?

The prototype that we have decided to create is a focused prototype, which as mentioned, is to test our application which utilises the heart rate/SpO2 subsystem created in prototype II. If we start with creating a simple app, this will provide insight as to how we can fix whatever issues that we are having, as opposed to creating a comprehensive prototype which is not possible given the current circumstances. This will be detailed in greater length in "Status Update." The analytical prototype was created from utilising code from a variety of Internet sources. The program that was used to create the app is called MIT App Inventor. The program utilises a "drag and drop" method of coding where one attaches multiple blocks together, thus illustrating a set of code. The main purpose of this software is to create applications that will be used on mobile devices. The applications can run on both the Android and iOS operating systems. We have decided to use App Inventor due to the fact that it is easy for newcomers to the computer programming scene to learn about the steps involved in computer programming and the software development process as a whole. Testing was performed by manually coding in low heart rate and SpO2 thresholds in the Arduino program. Then whenever the users placed their fingers on the sensor, there would be an automatic phone call made to the user's emergency contact list. The information that is being measured is the time interval between when the heart rate and SpO2 thresholds are reached and the subsequent phone call that is made.

2.2.1. Arduino Program Test

For this test, we have utilised code from multiple Internet sources. The objectives of this code are the following;

- Read the user's heart rate and SpO2 levels when one's finger is placed on the MAX30100. Then display these values on the serial monitor screen.

- Send the numerical values to the OP-Watch app, so they can be displayed on the app screen.

In order for this to work, a proper bluetooth connection had to be established which would send the values from the serial monitor to the OP-Watch app. Because the physical prototype remains the same, our current cost is $25.49. This is in the realm of the $100 budget, therefore satisfying one of the criteria; the device is cost-effective.

2.2.2. OP-Watch Application Test

The testing process is very simple, however there are multiple steps that must be taken before the application can be tested. In order to create a fully functional app, all the components must be grouped together in the code blocks of MIT App Inventor. Such components include the following;

- The heart rate and SpO2 levels will be displayed on the app which will allow the user to monitor their health status
- Emergency calling system that will alert the user's emergency contact list indicating that he/she is experiencing an overdose
- GPS tracking system that will display the longitude and latitude coordinates indicating the exact location of the user
- Sound function will also be implemented in the app to alert others nearby if the user has undergone an overdose

After the completion of combining these components, various tests will be conducted to assess the functionality of the app. Firstly, the heart rate and SpO2 levels will be tested to ensure the values are displayed in the app. This will be tested by the user placing their finger on the sensor to make sure the correct values are being read on the app. After this, the testing of the emergency calling system will take place. After this test is successful, we will make sure that the emergency calling system is operating properly.

In doing so, we will implement two options for the user to choose from. The first option, detailed in **Figure 1,** asks for the user to save a phone number from their emergency contact directory under the section "Store your emergency contact! Hit "'save"' when done." This function will allow the user to input a trustworthy emergency contact that will be responsive upon an overdose. For this test, one of the team members, "Ali Al-Zaidi", will input the contact name of one of the other team members' "Yazan Elmasry" to test if his number will be saved on the app. After the contact has been saved, there is a red button in the app entitled, "Test Call Emergency Contact". Using this feature we will be able to validate the emergency contact that was saved. This will call the emergency contact provided to ensure the calling feature works. During this call, on Ali's phone, a loud emergency sound will play for a duration of 20 seconds, thereby alerting others that he (in theory) has undergone an overdose. The second option, outlined in **Figure 2**, takes a much simpler and relient approach than the first in that there is a feature at the bottom of the screen entitled, "Flick Switch to Subscribe to Premium." When the user clicks the switch, it turns green, and a red button appears. This red button is called, "Test Call to Control Centre." In the event of an overdose, a phone call is made to a nearby control centre who will then forward the call to emergency services. In this way, the user will not have to worry if their loved one has been contacted or not; no matter the circumstances, the number will always reach a control centre. However the only caveat is that this is a subscription-based model, which implies that the user would have to pay an additional charge of $2.00 CAD for every 15 mins of calling.

The GPS tracking system encompasses an easier approach in the process of testing. For this test we will make sure that one of our team members has the device on hand. We will have the person walk around in front of their household. This will allow for the GPS coordinates to track the individual's exact location. This will be verified through having another team member type these coordinates into “Google Maps” which should correspond to the individual’s location.

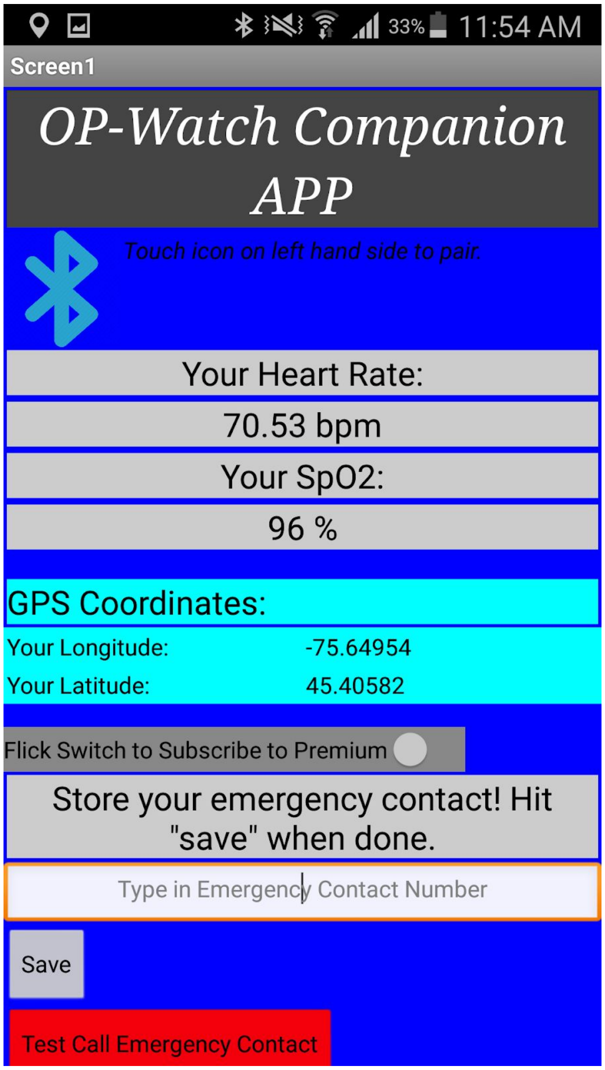
2.3. When is it happening?

The testing of the OP-Watch application will take place over the course of one day, where various individuals will come test the prototype and provide constructive feedback. In order for this prototype to go according to the test plan, our team will be deferential to the Gantt chart timeline outlined in “Project Deliverable E.” Abiding by the agreed-upon schedule will ensure that every team member is aware of the chronology for this prototype while also reducing the risk of failure that comes with this sequential order. The testing dependencies include creating the analytical prototype, which is expected to take one week of work. The testing will come to a complete stop after we have had ten people test the device for two minutes.

3. Prototype and Analysis:

The testing of the OP-Watch application was completed in consonance with the procedures detailed in the prototype test plan.

3.1 Prototyping Test Results:



OP-Watch Companion APP

Touch icon on left hand side to pair.

Your Heart Rate:

70.53 bpm

Your SpO2:

96 %

GPS Coordinates:

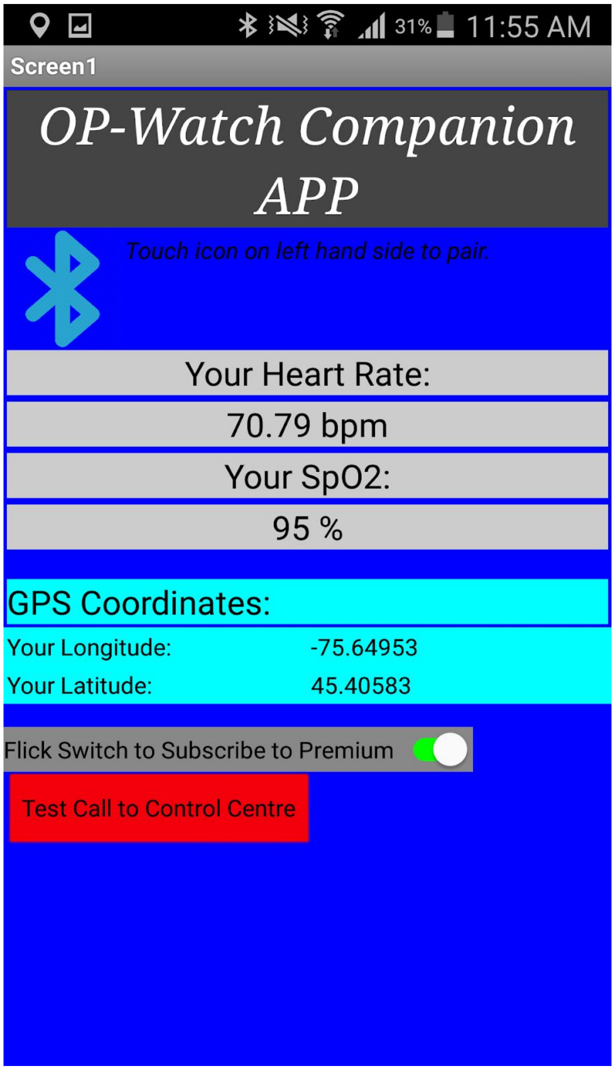
Your Longitude:	-75.64954
Your Latitude:	45.40582

Flick Switch to Subscribe to Premium ○

Store your emergency contact! Hit "save" when done.

Type in Emergency Contact Number

Figure 1: OP-Watch Main Screen (Option 1)



OP-Watch Companion APP

Touch icon on left hand side to pair.

Your Heart Rate:

70.79 bpm

Your SpO2:

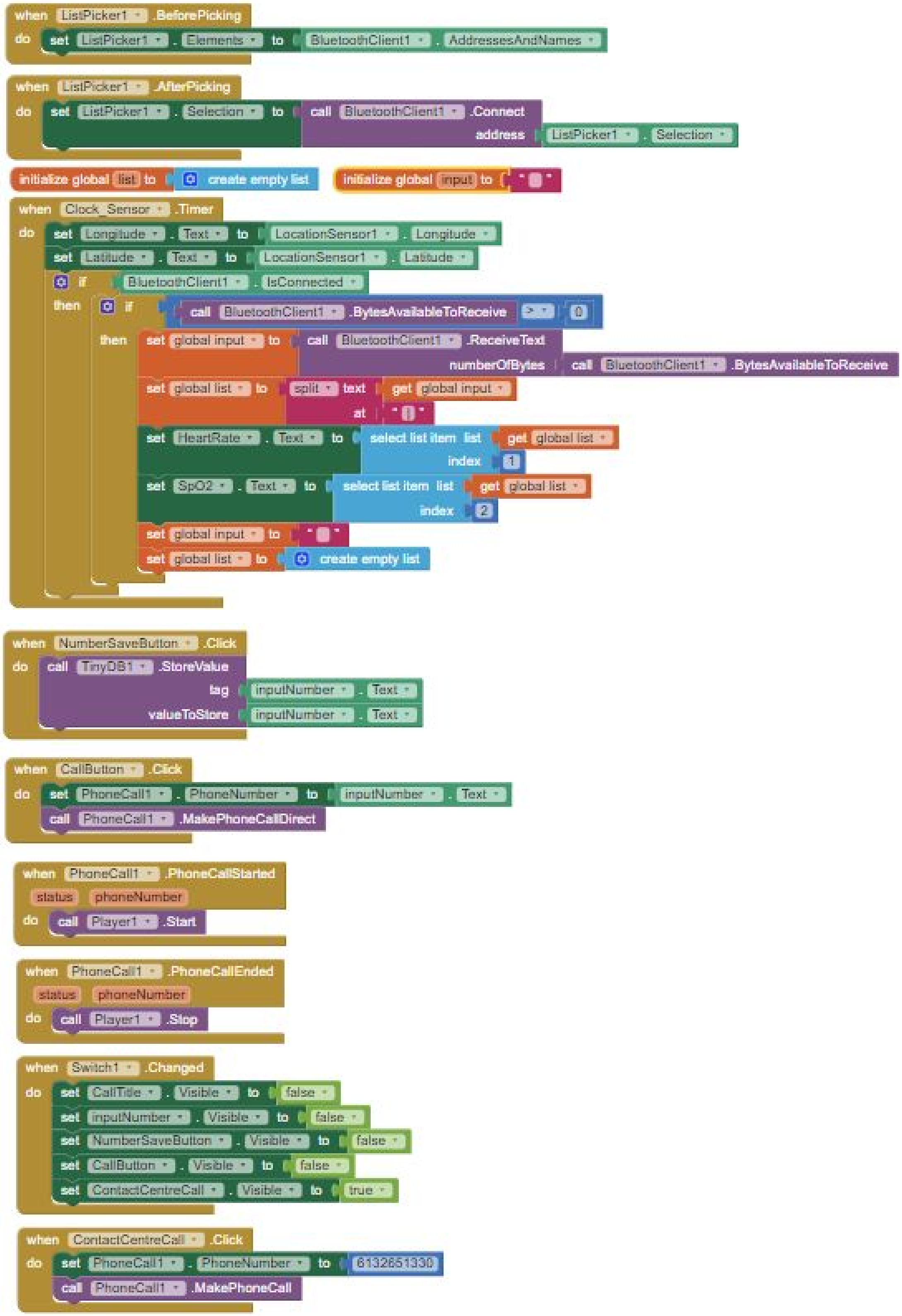
95 %

GPS Coordinates:

Your Longitude:	-75.64953
Your Latitude:	45.40583

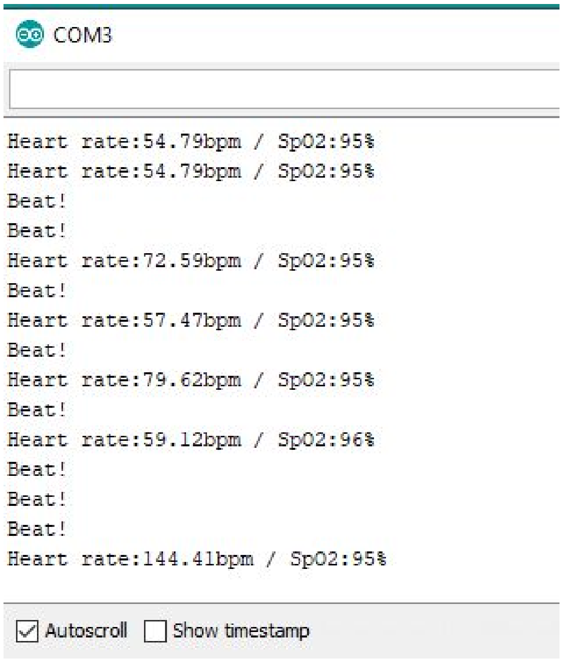
Flick Switch to Subscribe to Premium ●

Figure 2: OP-Watch Main Screen (Option 2)



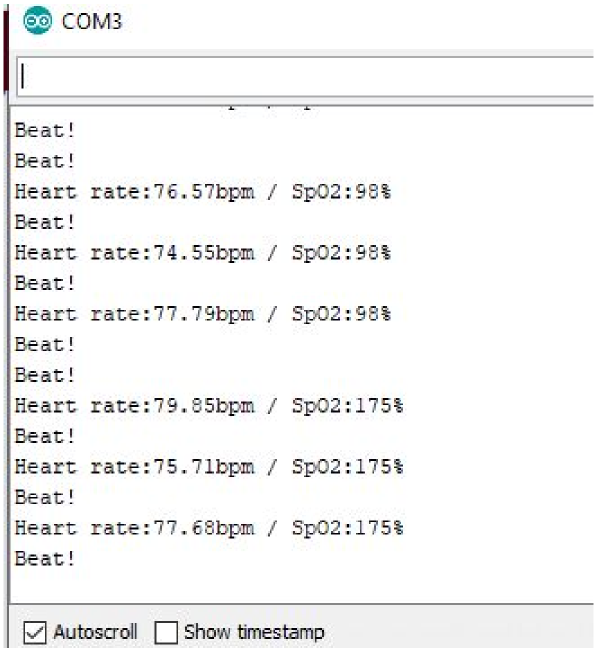
```
when ListPicker1 .BeforePicking
do  set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when ListPicker1 .AfterPicking
do  set ListPicker1 . Selection to call BluetoothClient1 .Connect
                                                    address ListPicker1 . Selection

initialize global list to create empty list
initialize global input to " "

when Clock_Sensor .Timer
do  set Longitude . Text to LocationSensor1 . Longitude
    set Latitude . Text to LocationSensor1 . Latitude
    if BluetoothClient1 . IsConnected
    then  if call BluetoothClient1 .BytesAvailableToReceive > 0
          then  set global input to call BluetoothClient1 .ReceiveText
                                   numberOfBytes call BluetoothClient1 .BytesAvailableToReceive
                set global list to split text get global input
                                         at "|"
                set HeartRate . Text to select list item list get global list
                                                         index 1
                set SpO2 . Text to select list item list get global list
                                                    index 2
                set global input to " "
                set global list to create empty list

when NumberSaveButton .Click
do  call TinyDB1 .StoreValue
                      tag inputNumber . Text
              valueToStore inputNumber . Text

when CallButton .Click
do  set PhoneCall1 . PhoneNumber to inputNumber . Text
    call PhoneCall1 .MakePhoneCallDirect

when PhoneCall1 .PhoneCallStarted
  status  phoneNumber
do  call Player1 .Start

when PhoneCall1 .PhoneCallEnded
  status  phoneNumber
do  call Player1 .Stop

when Switch1 .Changed
do  set CallTitle . Visible to false
    set inputNumber . Visible to false
    set NumberSaveButton . Visible to false
    set CallButton . Visible to false
    set ContactCentreCall . Visible to true

when ContactCentreCall .Click
do  set PhoneCall1 . PhoneNumber to 6132651330
    call PhoneCall1 .MakePhoneCall
```

Figure 3: MIT Inventor Code for the OP-Watch App



```
Heart rate:54.79bpm / SpO2:95%
Heart rate:54.79bpm / SpO2:95%
Beat!
Beat!
Heart rate:72.59bpm / SpO2:95%
Beat!
Heart rate:57.47bpm / SpO2:95%
Beat!
Heart rate:79.62bpm / SpO2:95%
Beat!
Heart rate:59.12bpm / SpO2:96%
Beat!
Beat!
Beat!
Heart rate:144.41bpm / SpO2:95%
```

Figure 4: Instance of Fluctuation in Heart Rate Levels



```
Beat!
Beat!
Heart rate:76.57bpm / SpO2:98%
Beat!
Heart rate:74.55bpm / SpO2:98%
Beat!
Heart rate:77.79bpm / SpO2:98%
Beat!
Beat!
Heart rate:79.85bpm / SpO2:175%
Beat!
Heart rate:75.71bpm / SpO2:175%
Beat!
Heart rate:77.68bpm / SpO2:175%
Beat!
```

Figure 5: Instance of Fluctuation in SpO2 Levels

4. Status of Prototype:

4.1 Final Update:

Unfortunately, access to the CEED facilities at the University of Ottawa ended abruptly after the COVID-19 pandemic near the end of the winter 2020 term, and work on a physical prototype came to an end. With team members unable to work together on tasks, productivity was greatly reduced. While this was a major and unique setback for our class, the team turned its attention to the app and worked hard to complete whatever possible under the circumstances.

In terms of hardware, the current prototype runs on a breadboard as there was no access to soldering equipment to make connections. The components currently working are the ESP32 board with built-in-Bluetooth, and the MAX30100 sensor. The OLED display intended to showcase the time and measured heart rate and SpO2 values was not connected nor working at the time of facility closures, and it was not possible to work on it at home as it requires parts such as resistors and pin headers. It is therefore not part of the finished work. The battery and vibrational motor are also not connected as the team intended to order them at the time when the facilities closed, but it was decided to cancel the plan as it would not have been able to assemble the battery and motor to the device. Therefore it is only the heart rate and SpO2 sensor working.

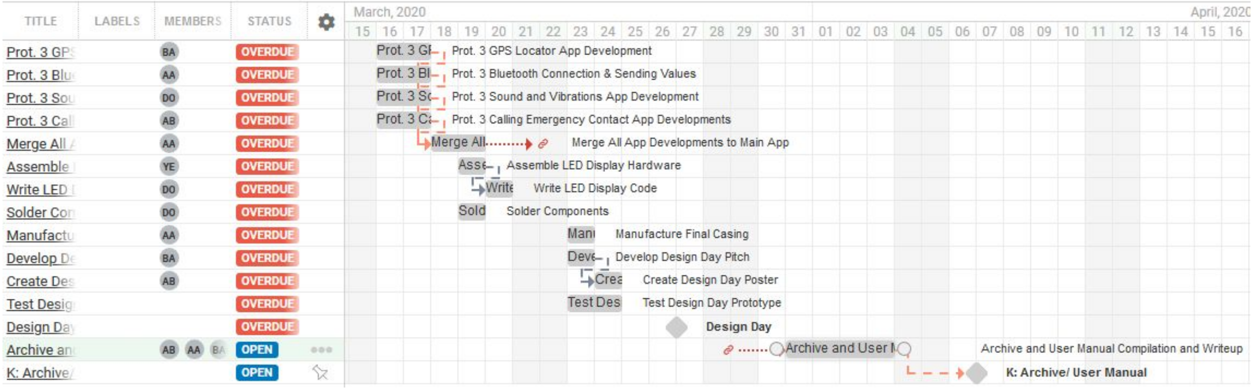
The ESP32 board is successfully sending values through Bluetooth signals, and the developed app is receiving and displaying the values. The app is also collecting coordinates from the host phone's GPS system, and displaying them on the user interface. In addition, we have developed and tested calling from the app successfully, as well as the noise and vibrations. These are the critical subsystems which we have put into one app. They are all working perfectly individually but they are not interconnected as a full system.

Initially the team's idea was to have all the devices contact a singular number, representing a call centre where administrative staff would collect data and send it to emergency aid. The client, however, mentioned during the five minute pitch that this could only work on a subscription model, and that this would not be possible. After facing the dilemma, as many engineers do, whether to prioritize highest possible reliability or lowest possible cost, it was decided to let the user decide what model they wish to opt-into. The developed app has a switch that, if disabled, allows the user to save a phone number which would be called in the case of an overdose. If enabled there is no longer an emergency contact number input value from the user. Instead the app calls the control centre's number in the case of an overdose detection. This feature as well as all those mentioned in this section are illustrated in the video found in the following link:

https://www.youtube.com/watch?v=SAfGZYoVknE

4.2 Planned Execution:

During the final stretch before Design Day, the team had agreed to physically meet every weekday for at least an hour, and weekends if necessary, to achieve a fully functioning prototype for Design Day. With this highly discouraged during the pandemic and facilities closed, as well as Design Day itself, it was no longer possible to achieve the goal, and the team lost its sense of direction and focus/clarity, having to adjust to new deadlines and expectations. Had events taken course as uncompleted tasks we would have aimed to complete include the assembly and coding of the LED display, soldering of the components, manufacturing of the final case, and the integration of the completed app subsystems into a fully functioning app. The following Gantt chart displays a day by day plan of how the team would have completed these tasks:



TITLE	LABELS	MEMBERS	STATUS
Prot. 3 GP		BA	OVERDUE
Prot. 3 Blu		AA	OVERDUE
Prot. 3 So		DO	OVERDUE
Prot. 3 Cal		AB	OVERDUE
Merge All		AA	OVERDUE
Assemble		YE	OVERDUE
Write LED		DO	OVERDUE
Solder Co		DO	OVERDUE
Manufactu		AA	OVERDUE
Develop D		BA	OVERDUE
Create De		AB	OVERDUE
Test Desig			OVERDUE
Design Da			OVERDUE
Archive an		AB AA BA	OPEN
K: Archive/			OPEN

March, 2020
April, 2020
15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16
Prot. 3 GPS Locator App Development
Prot. 3 Bluetooth Connection & Sending Values
Prot. 3 Sound and Vibrations App Development
Prot. 3 Calling Emergency Contact App Developments
Merge All App Developments to Main App
Assemble LED Display Hardware
Write LED Display Code
Solder Components
Manufacture Final Casing
Develop Design Day Pitch
Create Design Day Poster
Test Design Day Prototype
Design Day
Archive and User Manual Compilation and Writeup
K: Archive/ User Manual

Figure 6: Gantt Chart of 1.5 Weeks Before Design Day

It is important to note that if there is more than one task assigned in a time frame, they are assigned to different people. The work is split up between members for maximum efficiency. The first thing that was meant to be completed is the individual work of the team members on their assigned subsystem of the app, followed by the merger of these systems. Then the LED display is assembled with the rest of the hardware and everything is soldered in the most compact way possible. The display is then coded to display the time and measured values of heart rate and SpO2. The final casing is subsequently manufactured at the Brunsfield Centre from metal sheets with the refined process following the first prototype feedback, and the components are placed inside the casing, completing the Design Day prototype to be tested for a couple of days. An analytical model of the team's concept can be found in **Appendix 1**. At the same time the Design Day poster and pitch are made. The pitch is intended to be 2 to 3 minutes long and briefly summarizes the design process of the solution. After the completion of these tasks, the team would have been ready for Design Day on the 26th of March, 2020.

Considering the current pandemic, we were also unable to receive constructive feedback for our planned final prototype. If we had completed the final prototype with all the components as aforementioned, our team would showcase our physical prototype and functional app to potential clientele. By showing the clientele our final device and ready-made app, we would have provided them with a full demonstration on the device's usability, wearability, and the functionality of the watch's notable features as well as the app. Once we have given the clientele a full breakdown on our device, we would ask them to try on the device and test it for themselves to observe how they interact with the app's interface and the watch's comfort on the wrist. Based on this notion, the clientele would provide the team with feedback expressing their feelings and emotions towards the design of the watch. This would have given our team a better understanding on what aspects of our final design are exceptional and what features they may disagree with. Once the feedback has been received, our team would have gone through the comments carefully, to inspect for possible changes that could be made if this device were to be used by a real opioid user.

5. Conclusion

Throughout this report, our team has displayed the entire prototyping contingency test plan for prototype III, as well as the results and analysis of the device itself. The test results from this prototype indicate that the app is successful in that a call to an emergency contact is made when simulating the user's "opioid overdose." The emergency contact is able to observe the user's exact location and can contact emergency services if need be, thus displaying the accuracy of the implemented GPS tracker. However due to a myriad of tribulations, namely the COVID-19 outbreak, we were unable to garner and assess customer feedback for our watch. This was very unfortunate because it comprised the original test plan that we sought to undertake. Now that the prototype is "complete", our team will utilise the knowledge gained throughout the course of this project, and implement our design thinking process in the next and final deliverable entitled, "Project Deliverable K - User Manual."

References:

[1] [App Inventor Playground]. (2018, July 13). *Use the MAP component to show a map at your current GPS positions* [Video File]. Retrieved March 11, 2020, from https://www.youtube.com/watch?v=r9uVUps_gdg

[2] [CleanPNG]. (n.d.). *Pixel Art Logo* [Digital Image]. Retrieved March 26, 2020, from https://www.cleanpng.com/png-bluetooth-logo-computer-icons-symbol-web-2-blue-bl-6170956/

[3] Creating Easy Phone Call Application: Android App Tutorial. (n.d.). Retrieved March 18, 2020, from https://www.studytonight.com/android-app-using-mit-appinventor/creating-easy-call-app-using-app-inventor

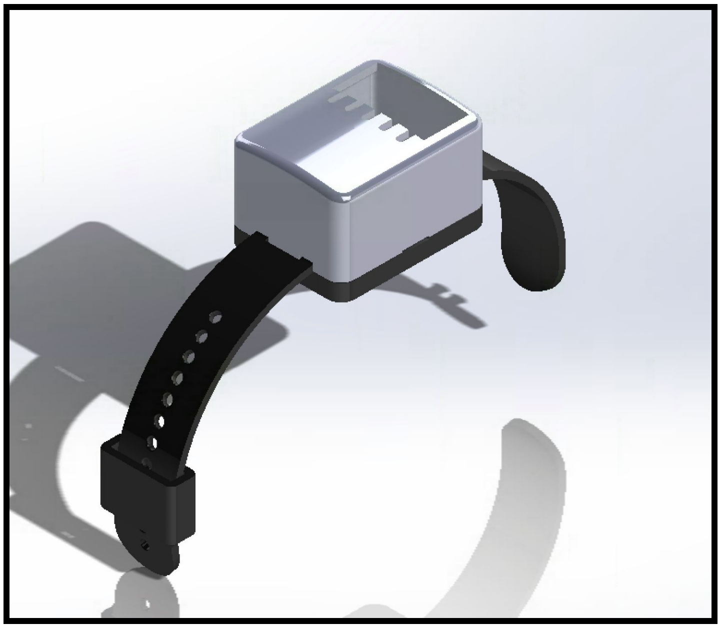
[4] [hundredvisionsguy]. (2017, Jan 5). *Adding Sounds to An App Inventor App* [Video File]. Retrieved March 12, 2020, from https://www.youtube.com/watch?v=-d7F285f1NA

[5] oxullo/Arduino-MAX30100. (2018, March 10). Retrieved March 6, 2020, from https://github.com/oxullo/Arduino-MAX30100

[6] [ZapSplat]. (n.d.). *Small emergency siren or alarm tone* [Audio File]. Retrieved March 19, 2020, from https://www.zapsplat.com/music/small-emergency-siren-or-alarm-tone/

[7] [Z-HUT]. (2016, Dec 11). *How To Receive Multiple Sensor Readings From Your ARDUINO Into Your MIT APP INVENTOR 2 APP* [Video File]. Retrieved March 15, 2020, from https://www.youtube.com/watch?v=5YovICoYCLg

Appendix:



Appendix 1: Focused model of Prototype III