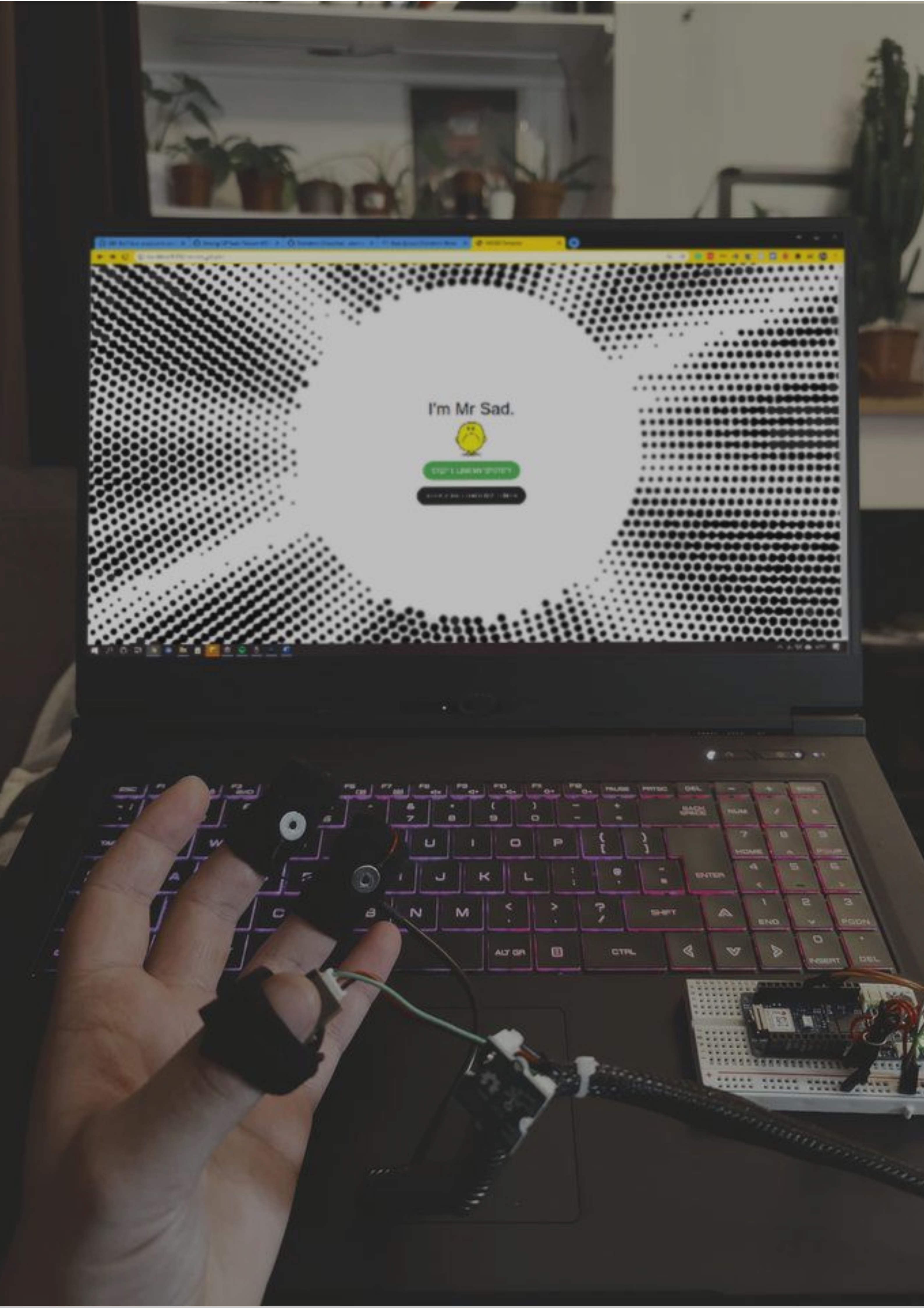
I'm Mr Sad.

Dyson School of Design Engineering | MEng Design Engineering

Mr Happy App – Project Report

Module code and Name	**DE4-SIOT Sensing & IoT**
Student CID	01348083
Assessment date	**4pm 14th Jan 2021**

Video demo URL:
https://www.higoralves.com/mr-happy-app or
https://vimeo.com/500192462

All files with instructions available at:
https://github.com/higorfelipe2/DE4_SIoT

Numbered links for referencing in this report:

1. **Raw data and synchronised data:**
   https://www.higoralves.com/mr-happy-app or

   https://thingspeak.com/channels/1277203 +
   https://thingspeak.com/channels/1279893 +
   https://thingspeak.com/channels/1280543

2. **Sensor Data Collection Code**
   https://github.com/higorfelipe2/DE4_SIoT/tree/main/2%20-%20sensor_data_to_thingspeak/Final

3. **Facial Expressions Data Collection Code**
   https://github.com/higorfelipe2/DE4_SIoT/tree/main/1%20-%20emotions_to_thingspeak/Data%20log%20before%20stream%20(1Hz)

4. **Web Application Code**
   https://github.com/higorfelipe2/DE4_SIoT/tree/main/5%20-%20web-app

5. **Data Analysis Code**
   https://github.com/higorfelipe2/DE4_SIoT/tree/main/3%20-%20data_analysis

1. Introduction and objectives

The extensive use of social networks has led to an increase in the spread of misinformation, due its propensity for spreading “farther, faster, deeper and more broadly” than real news (Vosoughi, Roy and Aral, 2018). This is in part due to miscommunication between information divulgers and receiver on social media platforms. Left unchecked, this is inherent to text-based media (Cornelius and Boos, 2003) and has resulted in a rise in political polarisation and social discord (Onook Oh, Kwon and Rao; Pew Research Center, 2014).

Current solutions being discussed by changemakers such as CEOs and politicians (e.g. Mark Zuckerberg (Facebook) and Jack Dorsey’s (Twitter) recent senate hearings (Yahoo Finance, 2020)) suggest changes that must be either made *by* or imposed *on* social media companies. Two schools of thoughts on reasoning, motivated reasoning (Kunda, 1990) and classical reasoning (Pennycook and Rand, 2018), both infer that presenting people with more information about a given topic increases the likelihood of them arriving at the truth.

This project explores the implementation one such information-based solution as a tool for promoting behaviour change. The proposed method involves capturing a user’s physiological response to some stimuli (watching videos in this case), in the form of photoplethysmography (PPG) and skin resistance; tagging this data with the user’s corresponding emotional response to it; then training a neural network with the physiological data as an input and the predicted emotions as an output. The results can be presented to the user, either its raw form to show the user their historical and current emotional response to some stimuli, or indirectly like in this project by showing the user what Spotify playlist best exemplifies their current emotional response to the stimuli.

2. Data Sources and sensing set-up

Data was collected using a PPG sensor to detect a pulse value, a galvanic skin response (GSR) sensor to detect skin resistance, and a webcam to capture pictures of faces. All in all, this encompassed two separate data collection processes, for three different time-series datasets. The Arduino set up for the physiological data is shown in Figure 1.

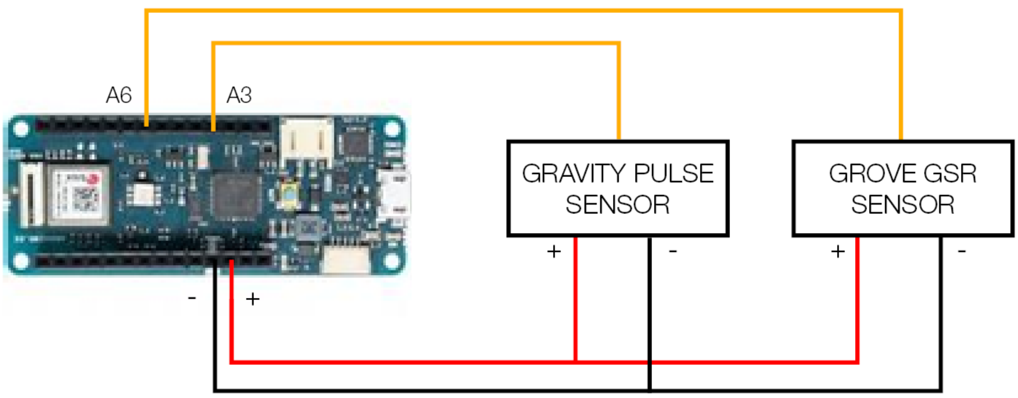

Figure 1: Arduino setup

It was initially planned to record heart rate in BPM as opposed to the raw pulse values, but the lack of hardware interrupt or useful standard libraries for the Arduino made this challenging. The complexity of the remainder of the project, including the fact that three time-series data streams had to be set up as opposed to the module's recommended two, meant that time must be allocated elsewhere. Having set up the sensor data collection, a python script was set up to take pictures using the laptop's webcam and retrieve emotions using the Azure Face API. This is described in more detail in the next section of this report.

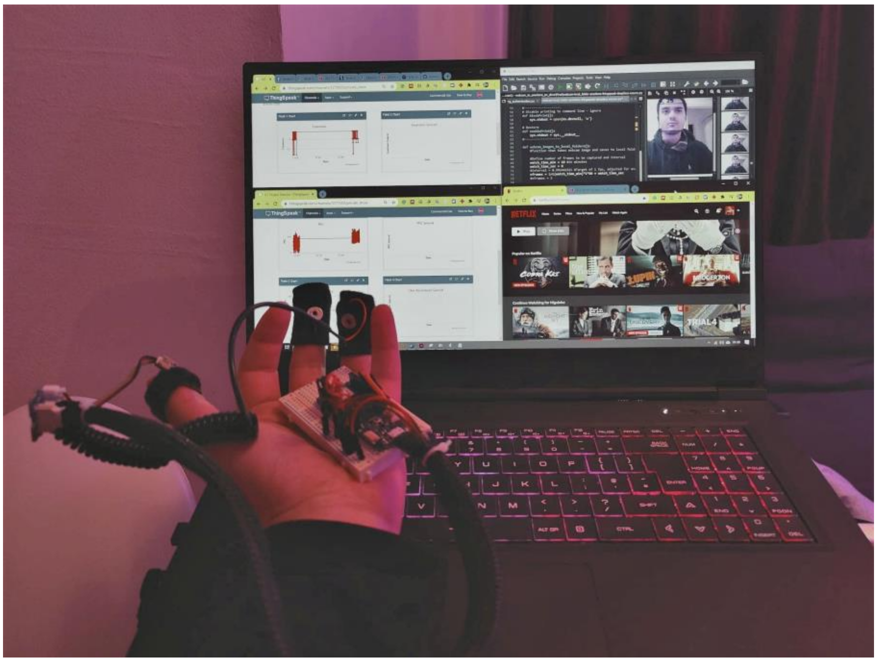

Figure 2: Overall sensing setup for data collection

Light and ambient temperature were maintained at similar levels to not affect the facial expression recognition and skin resistance, respectively. Using the setup on Figure 2, sensor and emotion data was collected at a frequency of 1Hz across 4 separate sessions of video watching (representative of 1 week activity), and then posted to two channels on ThingSpeak. The videos were chosen to elicit as many expressions and emotions as possible and included a variety of "Idiots in cars compilation", "Funny cats", "Sad short film" and half of "The Wolf of Wall Street". Both the raw historical and synchronised historical data are available from Link 1 on Page 1.

3. Data collection and storage process

During data collection and storage, the main consideration was the data rate limits imposed by the APIs being used. These two were the Azure Face API with a data rate limit of 3 images per minute (0.33 Hz), and the ThingSpeak API with a data rate limit of 1Hz. The sensor data was collected and published using the Arduino setup described above, in a single loop, with an interval of 1s added to avoid going above the rate limit. Initially, a similar single loop approach was used for the emotion data:

1. An image is taken.
2. Its corresponding expression/emotion is retrieved from Azure's Face API.
3. A 3s interval is added corresponding to Azure's data rate limit.
4. The emotion is converted to a number and posted to a ThingSpeak field.

This was not ideal as it would result in 3x fewer overall data points than the sensor data, since only data with matching timestamps could be used. This would amount to inefficient data collection. Whilst a premium can be paid to unlock a higher data rate for Azure's Face API if continuous streaming is required, this was prohibitively expensive, and a logging method was used instead:

1. All images are taken in 1Hz intervals and stored locally, with the timestamp as the filename.
2. Images are sent in 3s intervals to Azure's Face API to retrieve emotion data.
3. Emotions and current timestamps are stored into index-synchronised lists.
4. The emotions in the list are converted to numbers and posted to ThingSpeak at 1s intervals.

The code for this worked flawlessly and, during the longest data collection sitting, ran for a total of 90 minutes without error. This data logging process was at the cost of having to wait for 4x the data collection time (3x for Azure call+ 1x for ThingSpeak post). It was later noted that the ThingSpeak post loop could instead be within the Azure call loop to reduce this to just 3x the data collection time, but at that point the data had already been collected. A graphical representation of these loops is seen on Figure 3.

Since a redundancy was added during emotion retrieval to account for the user looking away or Azure's algorithm failing to detect an emotion, the resultant data was asynchronous. I.e., at this stage, there may have been some timestamps which had corresponding sensor data but no corresponding emotion data. To rectify this, the data was retrieved from ThingSpeak using Matlab and synchronised by removing non-common rows in a datetime table. As a result, the initial 6500 data points collected were filtered down to 5200 after synchronisation.

4. Basic characteristics of the systems set up and data

The characteristics of the system, up to the point where data can be directly fed into a web app, is shown in Figure 3 on the next page. The main characteristic is the difference in approach for sensor data collection and emotion data collection, where the latter requires data logging and three loops to ensure that the Azure Face API's data rate limit was not being exceeded.

5. Data Interaction/visualisation/actuation platform

The high complexity of the system discussed thus far had to be distilled to an interaction that was intuitive and highly useable. A web app with a simple UI but with all the functionality afforded by the large amount of data made available during the first half of the project was made using Python's Flask microframework and HTML. The resultant app uses a neural network fed with a live sensor data stream to predict users' current emotions. As an output, this predicted emotion is then used to display a corresponding character of the "Mr. Men" universe and used as a seed for making a Spotify Playlist relevant to the user's emotion. The front-end of this app is demonstrated in the supplementary demo video. In addition to the app's more intuitive user interface, more in-depth data visualisation is also available to the user via a link in the app through ThingSpeak iframes (Link 1). Figure 4 shows how variables are passed to the web app, and their corresponding UI elements.

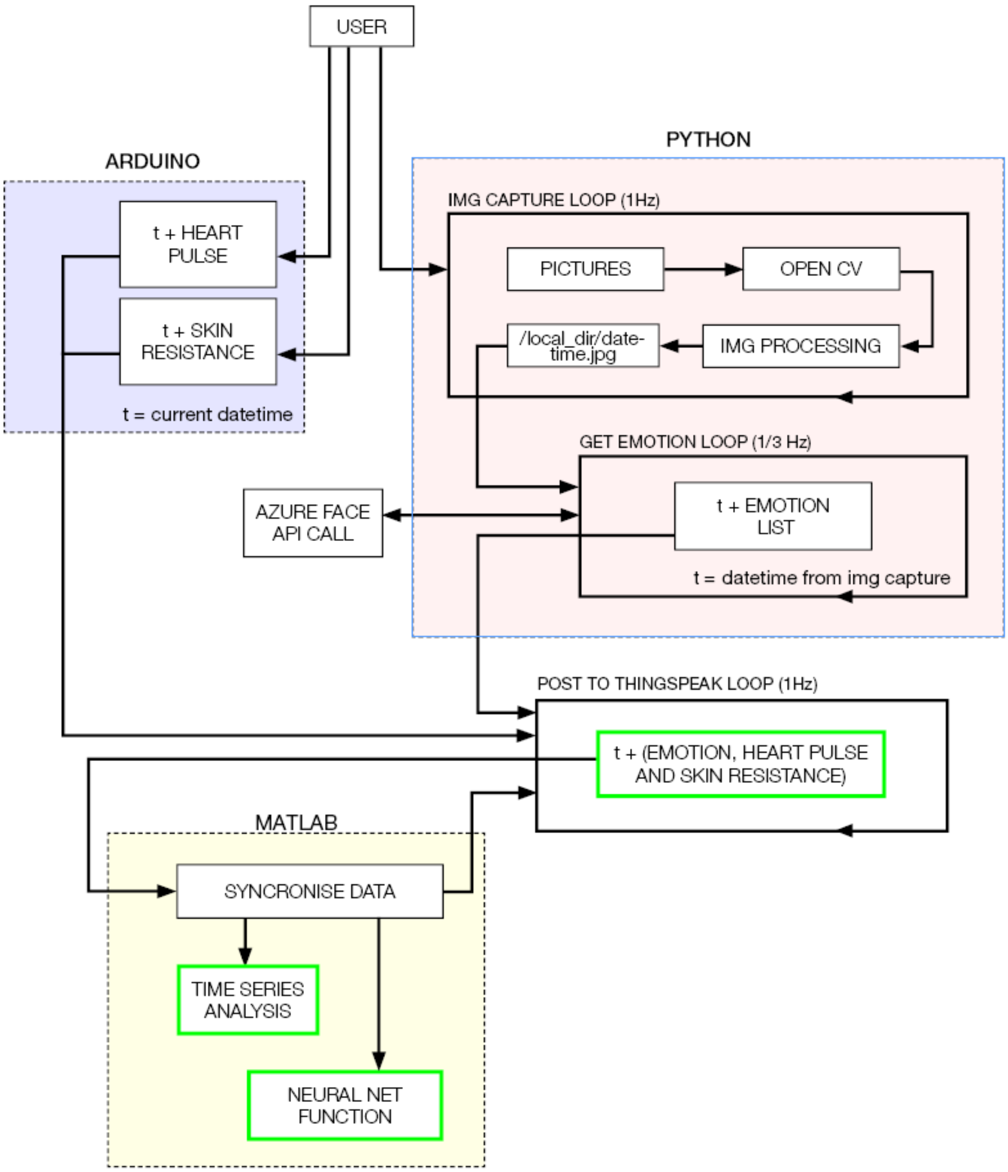

Figure 3: Characteristics of the end-to-end system set up.

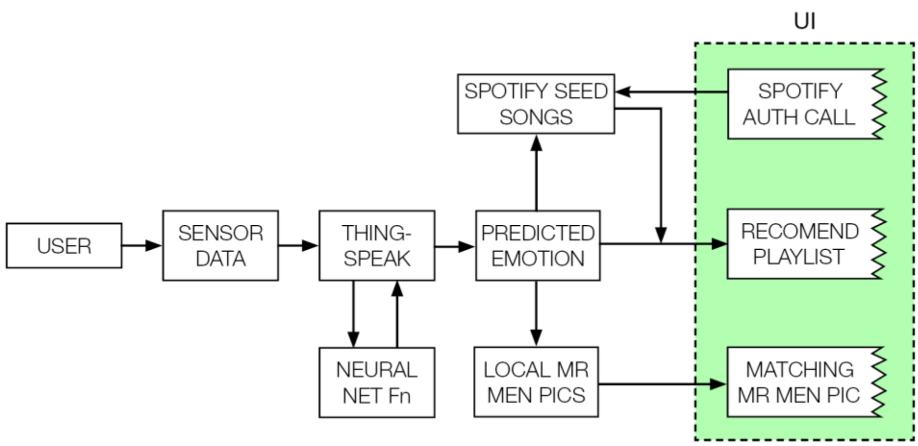

Figure 4: Web app visible and invisible layers. Note: sensor data is called again on refresh, updating predicted emotion and playlist recommendation.

6. Data analytics, inferences, and insights

The data synchronisation process resulted in 5221 data points across the three variables, PPG, skin resistance and emotion. It is important to note that for practical purposes such as being able to display them on a ThingSpeak graph, as well to be able to implement a neural network, the emotions were mapped to numerical values between -5 (angry) and 2 (happy). Whilst the emotions are not necessarily ordered as this would suggest, a rough approximation of order was intuited based on valence, where anger was determined to be the most negative emotion and happiness the most positive. It is expected that better neural network performance can be achieved by instead encoding this categorical data with 1-of-N encoding, wherein each of the 8 emotions can be expressed using a 1x8 array of zeroes and ones, with the column position of a single "1" dictating which emotion is expressed (e.g., Happy = [1 0 0 0 0 0 0 0], Sad = [0 1 0 0 0 0 0 0]). Other methods for improving the neural network performance are discussed further below.

Both the analysis of the dataset and the neural network created from this dataset were both achieved using normalised data where appropriate. I.e., the PPH and skin resistance values were normalised, but the numerical emotion values were not. The reasoning behind this is that emotion values would need to be inverse normalised at a later step to retrieve the corresponding emotion, and values in between the integers do not mean anything in practise due to the encoding method described above. After normalising the data, PPH and skin resistance outliers were removed, and then the corresponding emotions were removed from the MATLAB timetable. This is in contrast with removing outliers from all variables, which would result in all "non-zero" (non-neutral) emotions being removed due to the distribution of emotions (Figure 5) resulting in a mean of 0.18. Regardless of this practical purpose, there is no such thing as an outlier for emotions since a key assumption was made that the Azure Face API is accurate enough to be used for data tagging to begin with. In practise, the API was only 90% accurate during initial prototyping, but when compared to having to manually tag each emotion, this was deemed acceptable. A potential improvement is to log what certainty value the API reports for each returned expression and set a cut-off when the certainty value was anomalous (e.g., if the API prediction was not at least within 2 standard deviations of the mean certainty, do not log that emotion).

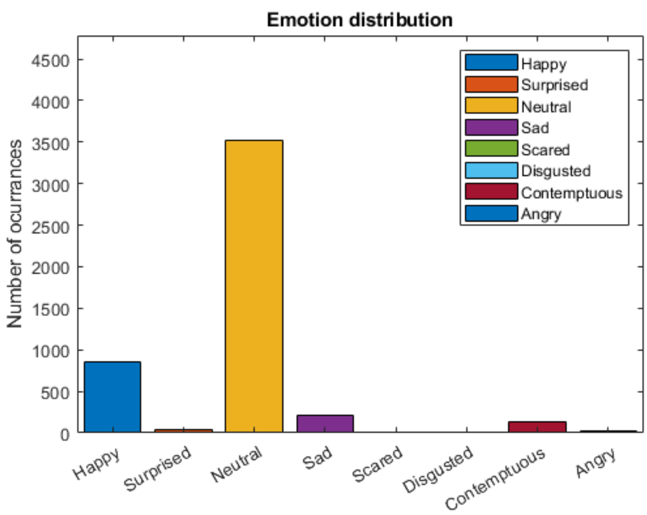

Figure 5: Emotions Histogram

A 2-layer feed-forward neural network containing 10 sigmoid hidden neurons and 10 linear output neurons was trained with a Levanberg-Marquadt back propagation algorithm using this dataset, split into 70-15-15 training-validation-testing groups. The network achieved an R value of 0.23 and a mean squared error (MSE) of 1.2. The performances of neural networks trained with just one input dataset, be it skin resistance or PPG, were considerably worse with R values averaging 0.11 and average MSE of 1.5, showing that whilst the performance overall was low, the use of two data streams was a good design decision. The error histogram in Figure 6 reinforces the fact that this would be unusable in a final packaged app; with instances way beyond this already high MSE resulting in a bad user experience. The time constraints of this project limited the implementation of known improvements. For example, as has been reported, one issue is converting the categorical emotion data to numerical data that the neural network can understand, and it is noted that 1-of-N instead of ordered encoding may lead to refinements. More fundamentally though, the dataset is flawed. As is seen in Figure 5, there is a significant skew towards a neutral expression, most likely explained by the transitory nature of facial expressions. To improve this more time would be needed for data collection, at least to the point where ~3000 instances of each emotion is present. Alternatively, more conventional methods for data tagging can be adopted. One such solution would involve doing sittings where only a video that elicits a known emotion from subjects is shown and tagging all PPG and skin resistance values outside of a baseline range with that emotion.

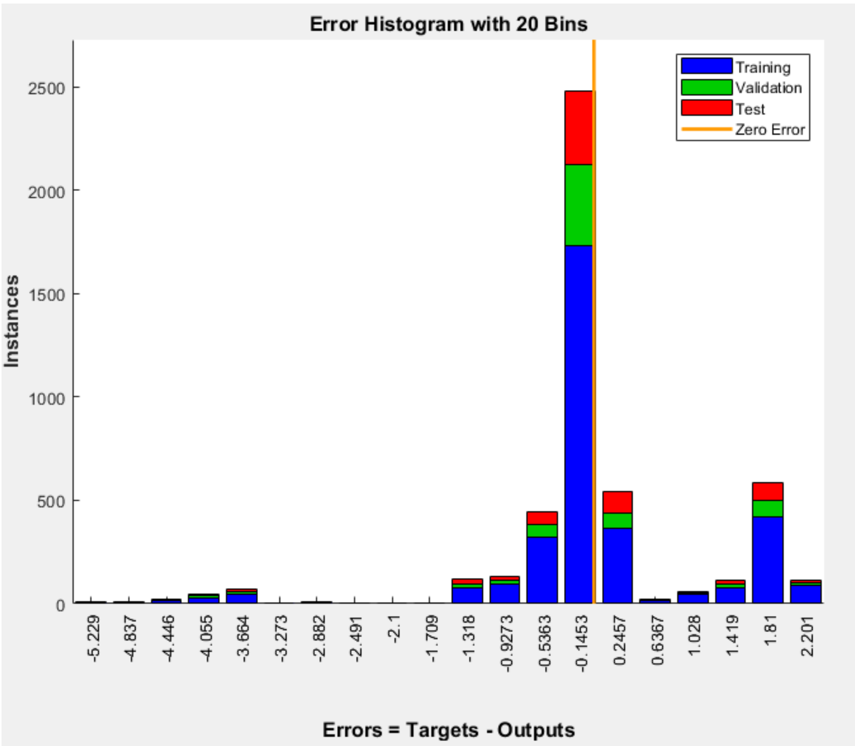

Figure 6: Neural Network Error Histogram

The windowed average of the sensor data and the windowed mode of the emotion data were plotted and analysed. As an example, for a window of width 60, samples from 1 to 61, 61 to 121 etc. were averaged. A few interesting phenomena were observed. First, due to the distribution of emotions shown in Figure 6, beyond a window of 150 values the mode graph of emotions was a flat line at zero (“neutral”). Considering the previous data processing which means sample and time in seconds are not one to one, this corresponds to a window of ~ 2-3 minutes. This reinforces the previous observation that emotions are transitory; only within two-minute windows can the modal emotion be anything but neutral. More interestingly, the correlation between mean PPG and mean skin resistance is at a maximum when the window was 240 samples wide, corresponding to ~5-7 minutes (Figure 7). Whilst care was taken during experimentation to condition the sensor for 10 minutes before each session to mitigate against sweat build-up and other factors contributing to variations in measurements, this suggests that this conditioning time was too low, and a systematic error was introduced as these sensors conditioned to the correct baseline. A method for investigating this phenomenon further, given more time, would be to perform a seasonality analysis of the dataset. Ideally, this seasonality would be filtered out in future data collection runs to account for other environmental conditions which may change the conditioning time. A similar analysis of moving standard deviation was also performed but no clear phenomenon was observed.

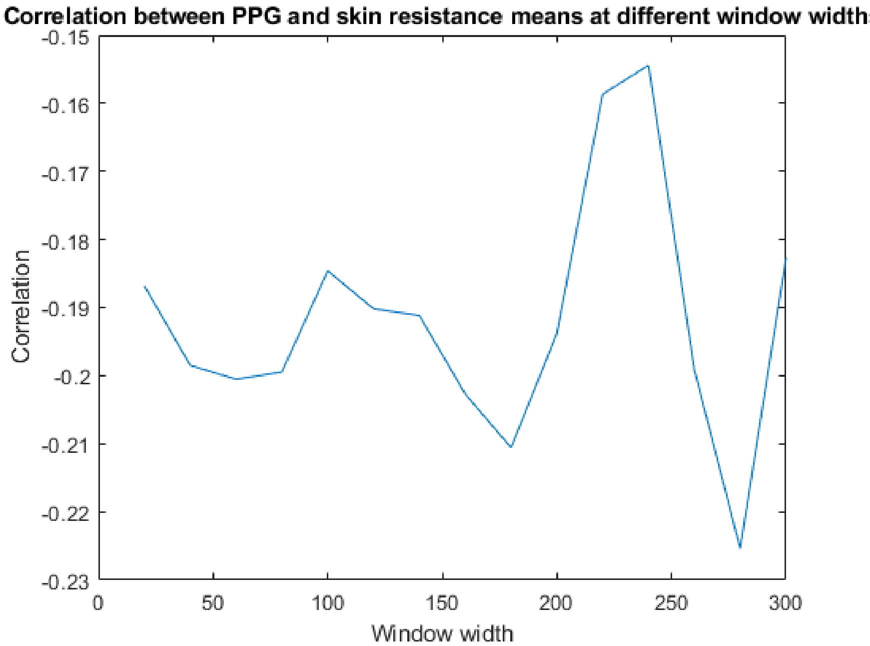

Figure 7: Correlations for PPG and skin resistance means at different window widths.

To conclude the data analysis, a series auto-correlation and a series cross-correlation analysis was performed. The autocorrelation of PPG, skin resistance and emotions were observed and overlaid. The autocorrelation of the first two datasets follows a more traditional curve, with a clear maximum at a lag of zero as expected, and a slowly receding distribution at lag values beyond this. The emotion auto-correlation, however, clearly shows more randomness and noise in the emotion dataset, suggesting that emotions are not necessarily correlated with each other. The auto-correlation graph (Figure 8) shows the required perfect correlation at zero lag, but even small lag values beyond this result in little correlation. The cross-correlation graph (Figure 9) confirms previous reporting, showing that the correlation is strongest between the PPG and skin resistance datasets. Interestingly though, the graph shows that the strongest correlation between either sensor data with the emotion data occurs at a lag of approximately 1200. This corresponds to ~19-20 minutes, which is longer than three of the four data collection sessions. This suggests a spurious correlation which does not equate to any causation and is therefore less useful in a predictive model. Note that this graph is normalised and only shows relative relationships between the three cross-correlations, the max values themselves are -0.116, -0.0936 and 0.0767 for PPG vs skin resistance, PPG vs emotions and skin resistance vs emotions, respectively.

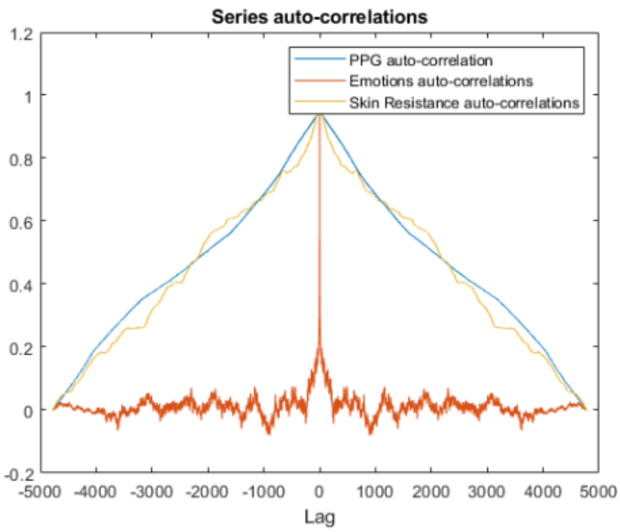

Figure 8: Autocorrelation for all data

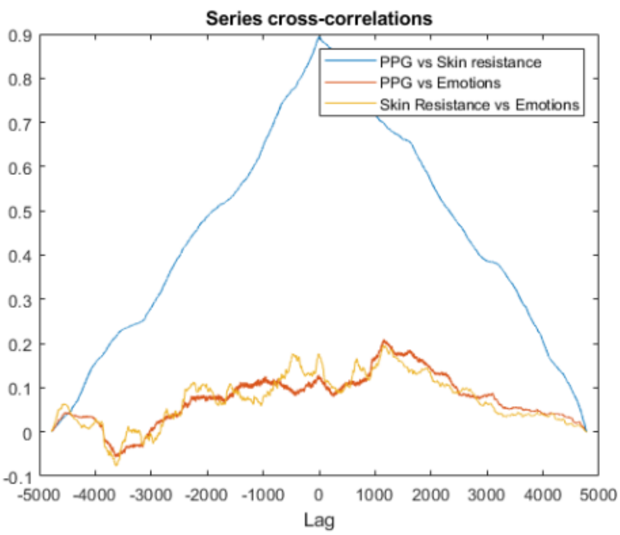

Figure 9: Cross-correlation for all data pairs

7. Discussions on the important aspects of the project

There are several project highlights which characterise the project and define the app's current functionality.

First, the rigour of implementation for the hardware component of the project is extensive, as was shown by its flawless functionality during data collection. This rigour also extends to the implementation of emotion data capture using a webcam, open CV, and Azure Face API, with sufficient redundancies put in place to ensure that interruptions, such as the subject covering or turning their face during data collection, did not throw up an error or misalign the emotion and its timestamp. Another example is the implementation of a more complex script which allowed the Azure Face API's 0.33Hz data limit to be bypassed by logging pictures and emotions locally before posting to ThingSpeak. The result of this rigour was a plug-and play data capture solution that allowed for as much data collection as possible within the project timeframe. It also means that this project can be continually explored, since all that is needed for improved functionality is more data and a refined neural network.

The implementation of a neural network to predict emotion is important for the app's functionality but is currently too inaccurate to be fully launched. Automated data tagging through facial expression detection, as was done in this project, is a feature that has not been fully explored in literature and is a good academic outcome for this project. That said, this contradicts with what may currently be best for the app. The availability of literature detailing other, non-automated methods for emotion tagging, would likely lend itself to the design of a more accurate emotion prediction model which would improve the app's functionality. An example of these non-automated methods of tagging involves, as described in Section 6, planning each data collection session around a specific video of known emotion, and then tagging all data from that session with that emotion. As has been reported, the neural network would also benefit from more extensive data collection that ensure all emotions are equally represented. Regardless, it is noted that using two physiological data inputs (or maybe more in future), was good practise and resulted in a more accurate neural network.

The web application itself is also successful in remaining simple and intuitive, masking the complexity of the back-end. That said, currently the app only works with development hardware and as is described in the next section of this report, there is potential for this functionality to be greatly improved.

8. Avenues for future work and potential impact

The potential for the outcomes of this app to both influence future work and to cause an impact can be split up into two categories: that which is an iteration and improvement upon the current app; and that which uses this project's core ideas and develops new things entirely.

For the first, future work can be conducted to improve the app's functionality. Chief among these is the potential to incorporate consumer, non-development hardware such as smartwatches which already have the sensors necessary to integrate with this app. This development would only require minor tweaks to the current app, mainly to do with how users' current PPG and skin resistance value would need to be stored. This method would likely differ depending on if it is an Apple Watch or an Android Wear device, with each having their own API ecosystems and servers enabling data storage and retrieval.

For the latter, there is potential academic impact in the proposal to do automated data tagging by using emotion prediction algorithms which have a known high accuracy. The case for this will only be made stronger as these algorithms improve.

Perhaps more importantly, there is potential to use all the above, to develop a user feedback system which tells social media users how much their social media habits are affecting them emotionally. This would need to:

1. Be integrated with currently used devices such as smartwatches as described above.

2. Benefit from advancements in automated data tagging which would allow for pre-trained emotion prediction models to be tailored to the user using transfer-learning.

This proposal would provide data that elicits behaviour change, helping to tackle the issues of increased social discord and political polarisation associated with social media usage, as described in the introduction.

9. References

Cornelius, C. and Boos, M. (2003). Enhancing Mutual Understanding in Synchronous Computer-Mediated Communication by Training. *Communication Research*, 30(2), pp.147–177.

Kunda, Z. (1990). The case for motivated reasoning. *Psychological Bulletin*, 108(3), pp.480–498.

Onook Oh, Kyounghee Hazel Kwon and Raghav Rao (2010). *An Exploration of Social Media in Extreme Events: Rumor Theory and Twitter during the Haiti Earthquake 2010.* [online] ResearchGate. Available at: https://www.researchgate.net/publication/221599216_An_Exploration_of_Social_Media_in_Extreme_Events_Rumor_Theory_and_Twitter_during_the_Haiti_Earthquake_2010 [Accessed 6 Dec. 2020].

Pennycook, G. and Rand, D.G. (2018). Lazy, not biased: Susceptibility to partisan fake news is better explained by lack of reasoning than by motivated reasoning. *Cognition*. [online] Available at: https://www.sciencedirect.com/science/article/pii/S001002771830163X [Accessed 31 Jan. 2019].

Pew Research Center - U.S. Politics & Policy. (2014). *Political Polarization in the American Public*. [online] Available at: https://www.pewresearch.org/politics/2014/06/12/political-polarization-in-the-american-public/.

Vosoughi, S., Roy, D. and Aral, S. (2018). The spread of true and false news online. *Science*, [online] 359(6380), pp.1146–1151. Available at: http://science.sciencemag.org/content/359/6380/1146 [Accessed 31 Jan. 2019].

Yahoo Finance (2020). *Twitter CEO Jack Dorsey and Facebook CEO Mark Zuckerberg testify on Capitol Hill - YouTube*. [online] www.youtube.com. Available at: https://www.youtube.com/watch?v=GSnLB_SafZc&ab_channel=YahooFinance [Accessed 6 Dec. 2020].