UHASSELT
KNOWLEDGE IN ACTION

Maastricht University

Faculteit Wetenschappen

School voor Informatietechnologie

master in de informatica

Masterthesis

Measuring and visualizing individual well-being throughout daily life

Mathias Gielen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :
Prof. dr. Gustavo Alberto ROVELO RUIZ

COPROMOTOR :
dr. Eva GEURTS

BEGELEIDER :
Mevrouw Anouk MICHIELS

De transnationale Universiteit Limburg is een uniek samenwerkingsverband van twee universiteiten in twee landen: de Universiteit Hasselt en Maastricht University.

UHASSELT
KNOWLEDGE IN ACTION

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

2023
2024

Faculteit Wetenschappen
School voor Informatietechnologie

master in de informatica

Masterthesis

Measuring and visualizing individual well-being throughout daily life

Mathias Gielen

Scriptie ingediend tot het behalen van de graad van master in de informatica

PROMOTOR :

Prof. dr. Gustavo Alberto ROVELO RUIZ

COPROMOTOR :

dr. Eva GEURTS

BEGELEIDER :

Mevrouw Anouk MICHIELS

Universiteit Hasselt

Masterproef voorgedragen tot het behalen van de graad van master in de informatica

Measuring and visualizing individual well-being throughout daily life

Auteur:

Gielen Mathias

Promotor:

Professor dr. Gustavo Rovelo Ruiz

Co-promotor:

dr. Eva Geurts

Begeleider(s):

ing. Anouk Michiels

Academiejaar 2023-2024

Acknowledgements

First and foremost, I want to thank Professor dr. Ruiz, dr. Eva Geurts and ing. Anouk Michiels for their patience and support in realizing this thesis. I am also deeply grateful to Professor Dr. Rovelo Ruiz for allowing me to choose this subject, thereby giving me the opportunity to apply my computer science knowledge to a field which genuinely interests me. I would also like to thank the entire teaching staff of UHasselt for this enriching experience, for the enhancement of my knowledge, and for the scientific insight that I have gained. Additionally, I want to thank my close friends for the emotional support and encouragement they gave me when things started to get stressful. They motivated me to successfully complete this thesis. Finally, I want to thank my parents for giving me the opportunity to achieve my Master's degree in Computer Science.

Abstract

Although well-being is receiving more attention, numerous challenges still persist in promoting well-being and adopting a healthy lifestyle. One way to get more insight into the well-being of an individual, is to measure the amount of stress this individual experiences and find out the reason why they experience it. Stress is something which many people experience throughout daily life, be it at work or somewhere else. Unfortunately, prolonged stress has significant consequences, as it is one of the main contributors of cardiovascular disease. The question is: how can stress be effectively measured in a non-obstructive manner, and how can other well-being parameters related to stress be visualized in a way that reveals potential links and correlations? During this thesis, a prototype system is created which aims to fulfil this goal.

One of the ways to measure stress is by analysing the physiological properties of the RR-intervals in an ECG signal, or in other words the characteristics of time intervals between successive heartbeats. This includes analysing the heart rate, the variance in time between successive heartbeats, the different frequencies in the heart rhythm and geometrical features by plotting subsequent RR-intervals in a phase-space. In an attempt to find patterns between these properties and stress, a kNN (K-nearest neighbours) classifier is applied to predict and differentiate mental stress, physical stress and no stress from one-minute intervals of heart data recordings.

In order to efficiently collect heart data, a Polar H10 chest band sensor is used. The processing unit inside the chest band automatically detects the RR-peaks from the raw ECG signal, removing the need for additional processing. In order to collect the data from the sensor, a mobile application is developed. This application establishes a Bluetooth connection with the sensor, and simultaneously allows the user to log activities and moods in order to submit contextual information and incrementally train the kNN model, and receive real-time stress notifications from the predictions made by the model.

Finally, a web application is developed in order to visualize the acquired data. A special timeline is created which allows for direct comparison of physiological signals, subjective feedback and stress predictions over a selected period of time in order to gain better insights about when and why the user has experienced stress. This timeline allows for dragging and zooming, increasing the granularity of data within a desired time frame. Additionally, a trends page is created to highlight general patterns, potentially indicating enhancements in both mental and physical well-being. To assess the effectiveness of the web page, a user study with 19 participants is conducted. Each participant used the web application to retrieve information for specific scenario's. They were instructed to apply the think-aloud principle while their actions were recorded through voice and screen recordings in order to analyse their reasoning during the usage of the application. After using the web application, the participant was asked to fill in a form in order to determine the strengths and weaknesses of the application and its features.

Contents

Chapter 1

Introduction

1.1 Problem statement

Stress levels, whether stemming from physical or mental stress, often play a significant role in the context of well-being. Scientific studies have shown that prolonged work-related stress is a risk-factor in cardio-metabolic disease, and contribute to workplace deaths in cases of people with and without pre-existing cardiovascular-related conditions [18]. Furthermore, according to the Belgian federal government, cardiovascular diseases (CVD) are responsible for roughly 20.7 percent of all registered deaths in 2020 [36], which makes it all the more interesting to look into this. An example of a risk group are industry workers, which are individuals highly subjected to both mental and physical stress. Examples of mental stress factors relating to industry workers are the difficulties and risks of operating complex machinery and the lack of social support, whereas physical stress is the difficulty of performing manual labor due to physical conditions such as Musculoskeletal Disorders (MSDs) or injury, or more simply put, discomfort or pain. A common example of a musculoskeletal disorder is back pain, which is often caused by lifting heavy load with improper technique. Tensions, spams and restricted movement caused by prolonged back pain can have drastic consequences to a worker's quality of life and further induce mental stress [8]. People can not only endanger themselves by working with these underlying health conditions, but also other workers should they experience a stroke or heart failure while performing tasks such as operating heavy machinery. In order to prevent a worst-case scenario such as an event of hyperventilation or even a heart attack, it is important to efficiently monitor the state of well-being of these individuals. But how can this be achieved in an efficient and non-obstructive manner?

1.2 Objective

As previously mentioned, cardiovascular conditions significantly impact an individual's mental and physical health. Therefore, it is crucial to thoroughly examine data related to heart health and other bodily signals associated with these conditions without forming an obstruction for the workers during their working hours. This can be achieved through the usage of wearable sensors such as the heart rate sensor and the accelerometer, which are typically integrated in wristband devices such as smartwatches. However, it is also necessary to assess subjective stress in order to gain better insight in what the exact causes of underlying health factors measured by the sensors might be. This could be parameters such as perceived work pressure and satisfaction, or the type of emotion felt at a certain moment. Thus, some sort of application is needed to enable the user to submit these parameters, preferably a mobile friendly application for efficiency reasons. This application should also allow both the individual and someone with medical expertise to monitor his/her physical and mental health status. Thus, the objective of this thesis is to study how such a system can be developed, which can be used to monitor health and stress-related physiological parameters unobtrusively through wearable sensors. Additionally, the system has to include a mobile-friendly way to gather subjective stress data, such as the perceived work pressure, directly from the user. To enable the monitoring of physical and mental health data provided by the mobile application, a dashboard is developed. This dashboard does not only present raw health data, but also contextualizes the information presented, illustrating connections between the data and the feedback given by the user. The dashboard also needs to offer explanations as to why the user felt stress as a certain point in time. This property is known as 'Intelligibility', allowing a clearer comprehension of the user's health patterns and stressors.

Chapter 2

Stress indicators

It is a known fact that stress can contribute to cardiovascular disease. Regardless, it is essential to briefly discuss some of the underlying biological factors to help understanding this phenomenon. The first one is *homeostasis.* In short, this term refers to the regulation of vital internal variables such as body temperature, pH value and blood sugar to maintain life. It consists of three components: a receptor, a control center and an effector. An example of homeostasis is the regulation of body temperature. This happens, for instance, when the brain instructs the muscle to shiver to generate heat if a significant drop in temperature is detected by thermoreception in the skin. This kind of adaptive process to achieve homeostasis by the control center, is referred to as *allostasis.* Each adaptation has a cost and contributes to the *allostatic load*, which is the physiological cost to adapting to new conditions or stressors whilst maintaining homeostasis over time. Both homeostatic and allostatic control regulated by the autonomic nervous system (ANS). This consists of two primary divisons: the sympathetic nervous system (SNS) and the parasympathetic nervous system (PNS). The SNS is often associated with the “fight or flight" response, preparing the body for physical activity and thus increasing heart rate by releasing cortisol into the bloodstream. The PNS on the other hand is responsible for vagal stimulation to promote digestion, relaxation for recovery and energy conservation, which causes a decrease in heart rate (HR). Thus, the ANS influences the activity of the sinus node, also known as the natural pacemaker of the heart. These mechanisms of increasing and decreasing physiological responses are also illustrative of allostatic regulation, which also means that the ANS is prone to the wear-and-tear effect allostatic load. Such a phenomenon may lead to a compromised ability to effectively adapt to stress, affecting homeostatic and allostatic processes and potentially contributing to the development of cardiovascular disease. [21] [4]

2.1 Heart rate & heart rate variability

The heart rate can be measured with an electrocardiogram (ECG), which detects the electrical signals produced by the heart when it beats. Figure 2.1 shows an example of a typical ECG signal. The highest peaks in the signal are called the R-waves, which indicate the electrical stimulus for the heart to contract when it pumps out blood. To clarify, heart rate stands for the amount of R-waves recorded within a minute, and is therefore expressed in beats per minute (bpm). A common example of a rise in heart rate is when physical exercise is performed. Breathing occurs faster to provide more oxygen to the blood, and oxygenated blood is pumped at a higher rate into the muscles that are in need of oxygen as fuel during the exercise. During rest, the heart rate goes back down in order to conserve energy. Analyzing heart rate during periods of rest is crucial, mainly due to a significant correlation identified between resting heart rate and the risk of cardiovascular disease and sudden cardiac death [33]. Heart rate variability, on the other hand, measures the time intervals between these consecutive heartbeats and can provide insights into the autonomic nervous system’s function. These time intervals are denoted as RR-intervals. As previously mentioned, the parasympathetic nervous system is responsible for inducing relaxation and recovery, which is paired with a decrease in heart rate. When it fails to do this, the heart rate will stay within the same range and therefore the variability in time between subsequent beats will be low. [24] It is noted that there exist interactions between the sympathetic and parasympathetic limbs. Given an activity that causes a raise heart rate due to sympathetic stimulation, the same amount of parasympathetic stimulation causes a greater change (reduction) change in heart rate. This phenomenon is known as *accentuated antagonism.* [42] Furthermore, sympathetic and parasympathetic activity both influence the intrinsic heart rate, and is effectively measured by applying pharmacologic blockade. The intrinsic heart rate is known to significantly decrease by age, indifferent from gender. [27] The effects of sympathetic and parasympathetic activity on the heart rate can be defined by the following formula:

$$HR = m \times n \times HR_0$$

where $m \geq 1$ is the sympathetic influence, $0 \leq n \leq 1$, and HR_0 the intrinsic heart rate. [27] While heart rate is a valuable indicator for assessing autonomic tone, it lacks the ability to differentiate between sympathetic and parasympathetic influences independently. For example, a heart rate of 100 at a given time can either be due to a rise in sympathetic activity after a restful state or an increase in parasympathetic activity after exercise. To address this limitation, HRV can provide valuable insights into the stimulation of the SNS and the PNS.

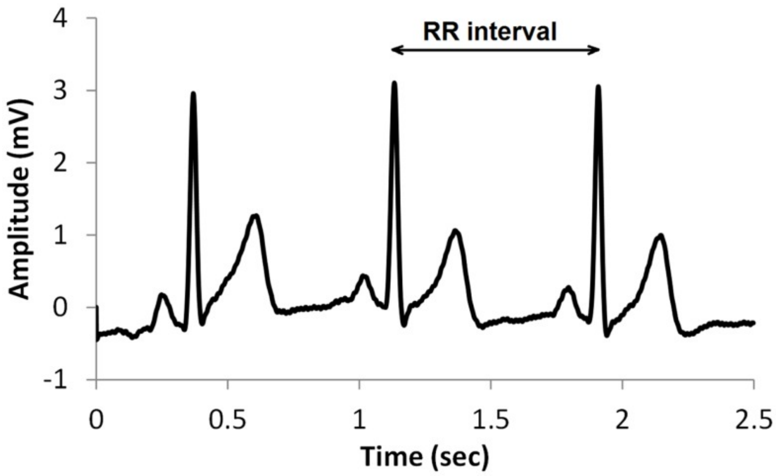


Figure 2.1: Example of a series of QRS complexes in an ECG signal [10]

2.1.1 RMSSD

There are several approaches to assess heart rate variability (HRV). One method involves time-domain analysis, which examines changes in the ECG signal over time. One time-domain method to determine the HRV is to calculate the root mean square of successive differences (RMSSD). It is a time-domain measure used to assess parasympathetic (vagal) activity, as it describes short-term variations within the heart rate data. Therefore, it is a useful tool when extracting data from sensors that provide time-series data. The RMSSD is calculated as follows:

$$\text{RMSSD} = \sqrt{\tfrac{1}{N-1} \sum_{i=1}^{N-1} (RR_{i+1} - RR_i)^2}$$

Reduced HRV is often linked to stress and anxiety, indicating a decrease in parasympathetic, relaxation-promoting activity. It is usually calculated within a short time interval, ranging from thirty seconds to five minutes. [24] In the formula above, N stands for the total amount of successive heart beats recorded within a given time frame and its value can therefore vary. To clarify: let's take a time interval of one minute. In a hypothetical case in which each RR-interval is roughly equal to one second, 60 beats would be registered within one minute and N would be equal to 60. However, 120 beats of roughly 500 milliseconds would also add up to one minute. So, given a series of intervals as shown in Figure 2.2, the HRV is calculated from each block with RR-intervals adding up to roughly one minute.

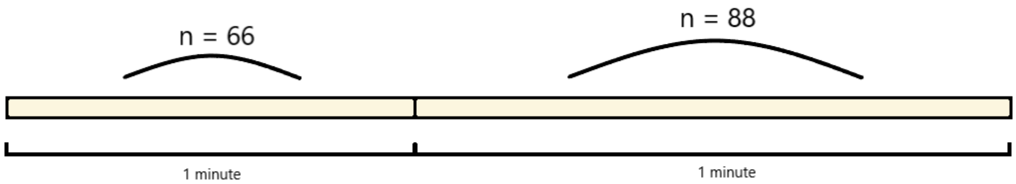


Figure 2.2: Segmenting RRI sample data into blocks of one-minute HRV

In situations where stressful activities can arise unexpectedly, it is beneficial to have more detailed HRV information. For example, if an individual experiences a stressful event, it is important to know when exactly stress occurred. From various time windows throughout the day, HRV calculations can be obtained to determine how the individual's parasympathetic nervous system is responding to stressors. To obtain the RMSSD at any given moment, a sliding window approach can be applied to a series of subsequent RR-intervals.

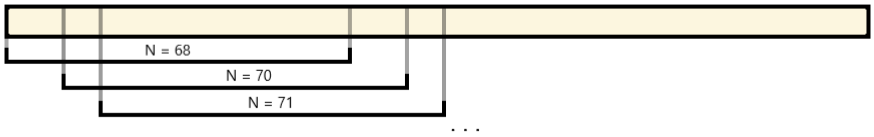


Figure 2.3: Sliding one-minute window

As explained earlier, each window can have a varying amount of RR-intervals adding up to roughly one minute. In order to 'slide' window as shown in Figure 2.3, the following RR-intervals are added to the front of a buffer until the sum of each RR-interval is equal to or exceeds one minute. If the total amount of RR-intervals in the buffer exceeds one minute, elements in the back of the buffer are removed until the sum of RR-intervals is less or equal than one minute. A `deque` (double-ended queue) is a useful data structure for this process as it allows for insertions and deletions on both sides in constant time. The initial HRV calculation requires a wait period equal to the chosen time window. Since it is necessary to accumulate a set of N RR-intervals first, immediate HRV calculations are not possible.

2.1.2 HF and LF power

Another method is frequency-domain analysis, which explores the various frequency bands present in a sequence of RR-intervals. The high-frequency band ranges from 0.15 to 0.4 Hz, while the low-frequency band ranges from 0.04 and 0.15 Hz. The power of a frequency band is the portion of waves from that band within a signal. Lower HF power has been shown to be a result of a decrease in parasympathetic activity. Thus, the HF band is known as an indicator for parasympathetic activity, and is therefore highly correlated with the RMSSD time-domain measure. Furthermore, it is also referred to as the "respiratory band" due to its correlation with the respiratory cycle. [21] [24]. LF power is influenced by both sympathetic and parasympathetic activity, and is known to be associated with baroreceptor activity. Baroreceptors are stretch-sensitive receptors located within the blood vessels and arteries. When a change in blood pressure occurs, these receptors sense a change in vessel wall stretch and send signals to the ANS to adjust heart rate in order to achieve homeostasis. [44] [24] [32]. Because ECG sensors provide time-series data, a method to convert data from time-domain to frequency-domain is needed in order to obtain the frequencies. Figure 2.4 explains how the different frequencies can be obtained from a time-domain sample:

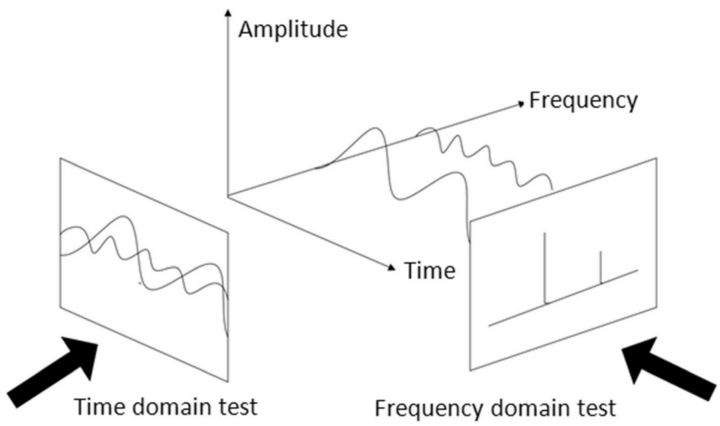


Figure 2.4: Time-domain and corresponding frequency domain [37]

The conversion from time-domain data to frequency-domain data can be achieved by applying the Fourier Transformation, which analyzes the frequencies present within a continuous function. [47]

$$\mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t)\, e^{-i2\pi ft}\, dt$$

In the case of ECG sensors, input function $f(t)$ would be the analog voltage signal measured at time t. However, to peform HRV analysis, we only need discrete RRI data and not continuous ECG data. RRI data can be derived from raw ECG data by applying signal processing techniques such as template matching, but most wearable sensors perform analog-to-digital conversions on their own and provide sequences of RRI as output. To perform frequency-analysis on this discrete data, the Discrete Fourier Transformation [45] can be applied to transform a series of real or complex numbers from the time domain into the frequency domain. These series of numbers form the digital signal x. The equation for the Discrete Fourier Transformation below can be solved in $\mathcal{O}(n \log n)$ time using the Fast Fourier Transformation algorithm [46].

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi kn/N}$$

Finally, the power spectral density (PSD) of the converted signal is obtained, which describes the power of each unit of frequency within a signal in watts per hertz (W/HZ). [48] $S(f_k)$ represents the power of frequency k. Thus, for the high-frequency band, we are interested in the summation of power of k-values between 0.15 and 0.4.

$$S(f_k) = \sum_{n=0}^{N-1} R[n] \cdot e^{-j2\pi f_k n}$$

Figure 2.5 shows an example of the frequency powers present within a raw analog ECG-signal. Other spectral density estimation methods include the Periodogram [**?**] and its improvement, Welch's method [49], which involve applying FFT to different, potentially overlapping segments of the same signal and averaging the obtained results from each frequency.

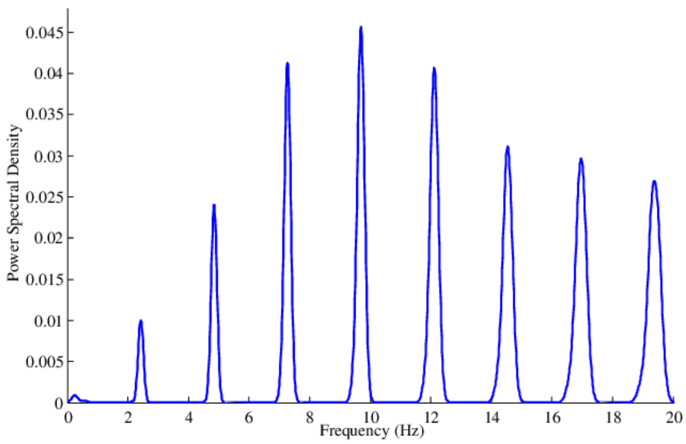


Figure 2.5: Power per frequency [23]

2.1.3 Poincaré plot

The Poincaré plot is an alternative, nonlinear method for assessing HRV using a scatterplot and geometrical analysis. In this approach, the series of RR intervals (RRI) is visualized in a phase-space. Each RRI is graphed against its preceding RRI, creating a scatterplot where each successive pair of RR intervals represents a point in the plot. By adjusting the plot to an ellipse, two parameters can be derived: SD1 (Standard Deviation 1) and SD2 (Standard Deviation 2). SD1 corresponds to the minor axis of the ellipse, while SD2 stands for the long-term RRI-variability which corresponds to the major axis as depicted in Figure 2.6

The standard deviations are calculated as follows:

$$SD1 = \sqrt{\frac{\sum_{i=1}^{N}(RR_i - \bar{RR})^2}{N}} \qquad SD2 = \sqrt{\frac{\sum_{i=1}^{N}(RR_i - RR_{i-1})^2}{2(N-1)}}$$

An advantage of the Poincaré plot over other metrics is that can visualize the balance between the two branches of the autonomic nervous system. SD1 is used to measure beat-to-beat variability, reflecting parasympathetic activity, while SD2 represents overall HRV or sympathetic and parasympathetic modulation combined. A stretched-out shape indicates a balance between sympathetic and parasympathetic activity, while a compressed shape may indicate dominance of the sympathetic branch. Standard deviation values are said to be bigger in normal cases than in disease cases. [38]. To conclude this summary of HRV calculation methods, it is noted that there is no single "gold standard" metric for HRV analysis, and the choice of a specific metric depends on the intended clinical objectives.

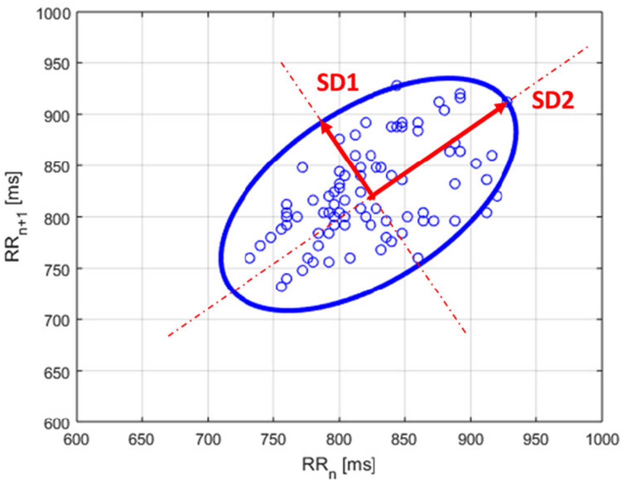


Figure 2.6: Poincaré plot example [19]

2.1.4 Implications

The nature of calculations within a short time interval such as one minute may pose some problems, particularly when individuals transition from a resting state to intense physical activity. During such transitions, both low and high RRI values may coexist within the calculation window. This disparity can potentially cause the calculated variance to increase tremendously, which does not accurately reflect parasympathetic activity. It is of no use to analyse these instances of high HRV, since it is only of interest to analyse when HRV drops below baseline during and after a stressful event. A problem that might arise when reading RRI values, is the introduction of artifacts. These are changes in the electrical ECG signal that are not caused by cardiac activity, and usually occur when the ECG is susceptible to movement during measurements, or when it is not attached properly to the skin. An artifact can, for example, cause an RRI to be three seconds. Under normal circumstances, this would indicate heart failure. An RRI value of three seconds rarely appears, so it is considered as an outlier. Therefore, the utilization of computational techniques for detecting and filtering outliers in a sequence of RRI induced by artifacts is needed. A common way to detect outliers is to calculate the Z-score of an RRI, which indicates how many standard deviations σ it lies from the mean RRI μ. An RRI with a Z-score greater than +3 or less than -3 is considered an outlier, as shown in Figure 2.9, and should be replaced. One of the ways to replace outliers is by applying linear interpolation. Suppose the outlier RRI along with its corresponding timestamp is denoted as $(x_{\text{outlier}}, y_{\text{outlier}})$, pick two neighbouring points (x_1, y_1) and (x_2, y_2). To obtain the interpolated data point $(x_{\text{interpolated}}, y_{\text{interpolated}})$, the following calculation is applied:

$$x_{\text{interpolated}} = x_{\text{outlier}} \qquad\qquad y_{\text{interpolated}} = y_1 + \frac{x_{\text{outlier}} - x_1}{x_2 - x_1} \times (y_2 - y_1)$$

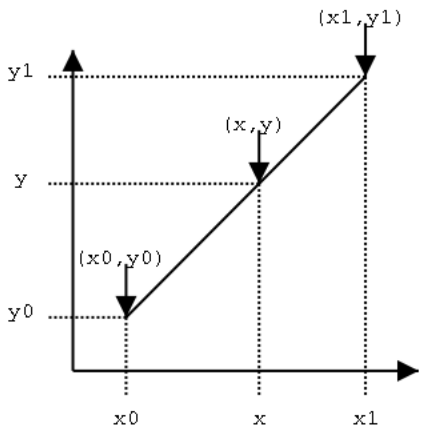


Figure 2.7: Linear interpolation [41]

The last thing to consider is ectopic beats. These are small irregularities in the heart rhythm such as having a faster heartbeat or skipping a beat. For example, the heart may have an increased pace after the consumption of caffeine. This is a natural response from the heart to caffeine, but it doesn't necessarily indicate heart disease and is unrelated to ANS activity. [50] Therefore, ectopic beats should also not be held into account when calculating HRV. An algorithm to remove ectopic beats is described by Kaçar, Murat et al. (2000), also known als the "Malik rule" [17]. The refined series of interbeat intervals from which artifacts are removed and therefore suitable for HRV calculation in practice, are called NN-intervals. In summary, ECG readings should be as little as possible contaminated with outliers and ectopic beats to be suited for analysis. To conclude this section, it is revealed how a low HRV can indicate problems with the ANS. However, it is important to be cautious and not take incorrect and/or premature conclusions conclusions solely based on HR and HRV measurements. Additional medical conditions and individual differences can also contribute to a change in HRV, so it should not used as a standalone metric.

2.2 Respiration and skin conduction

During inhalation, pressure inside the chest rises. This causes a drop in cardiac output, leading to a drop in arterial pressure. This change is detected by the baroreceptors, which trigger an increase in heart rate to maintain cardiovascular stability. During exhalation, heart rate slows back down. This phenomenon is called Respiratory sinus arrhythmia (RSA). Literature shows that a controlled, slow paced breathing pattern does not only provide cardiovascular improvements and exercise benefits, but also provides cognitive improvements and benefits handling stress and anxiety. [32] Therefore, respiration is an extra valuable measurement for gaining insight into the physical and mental aspects of well being. In this thesis, respiratory analysis is limited to measuring and analysing the respiratory rate, which is the amount of inhalations recorded within one minute. The previous section mentioned that the high-frequency band, which reflects parasympathetic stimuli and ranges from 0.15 to 0.4 Hz, is correlated with the respiratory cycle. By using spectral density analysis, it is possible to estimate the breathing frequency from a specific window of RR-intervals. How these calculations will be performed will be explained in later sections.

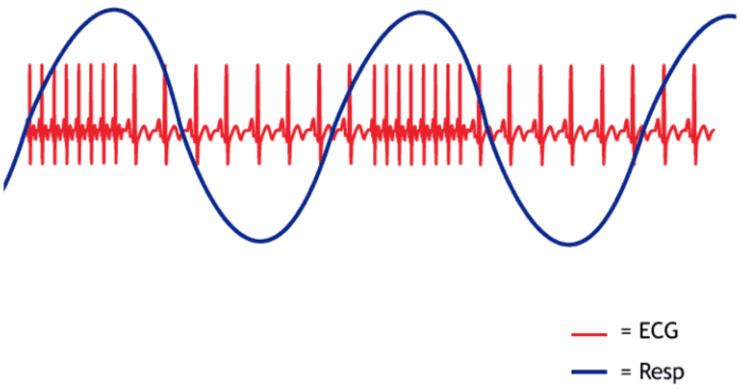


Figure 2.8: Relation between inhalation and heart frequency [25]

Another physiological marker of stress is electrodermal activity (EDA), which is the level of electrical conductivity of the skin measured in voltage. This metric is influenced by the amount of sweat the skin produces, and serves as a marker for sympathetic nervous system activity which is linked to stress, arousal and various other emotions. [7] Similar to HRV analysis, it is beneficial to contextualize these metrics and provide visualizations to observe their variations over time, deviations from baseline levels, and alterations in response to stressful events.

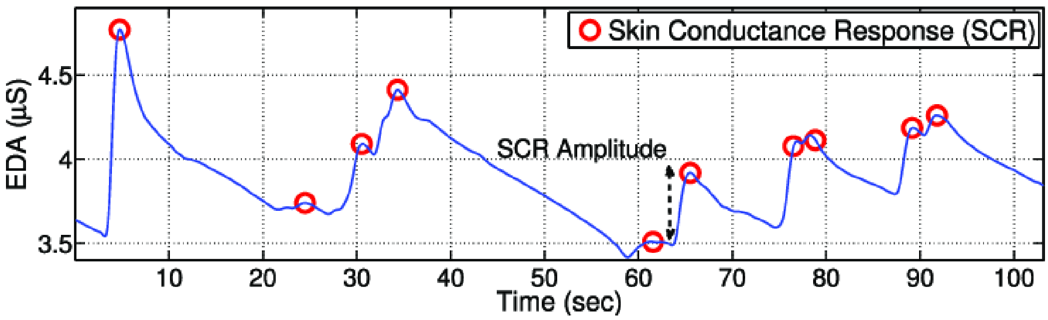


Figure 2.9: Example of an EDA signal [6]

2.3 Stressful events

During daily events such as being the work floor, there can be various factors that contribute to stress. Some jobs, for example, can be more mentally or physically demanding than others and the ability to meet those demands can differ from individual to individual. Furthermore, factors regarding the personal life of a worker such as relationships, social status and general well-being also contributes to how much subjective workplace stress a worker experiences. This was proven in a study by Low et al. (2018) [21], which established correlations between HRV and stress, intention to quit, relational tension and age. From an employer's perspective, it can then essential to know which aspects of a job are potentially stressful for a worker. In this thesis, stressful events are divided into two main categories.

2.3.1 Mental stress

This category refers to events in which stress is associated by negative emotion and caused by activities that do not necessarily require physical labour. Examples of such activities could be office work, doing intensive cognitive work, potentially being under the pressure of deadlines. An example regarding students might be anticipating or giving an important presentation in front of people, which is something that students at Hasselt University are all too common with. A more subtle example contributing to chronic mental stress is the use of cell phones, instant messaging programs, and social networking sites. This creates an expectation for people to be constantly reachable and to address matters outside of normal office hours, thereby blurring the line between personal and work time for many individuals. Consequently, individuals may feel overwhelmed, causing them to always be on the edge of being stressed because they rarely get any break from expecting to receive messages all day long. It becomes evident that in many cases, the perceived notion of stress can be likened to feelings of frustration, anxiety, and a variety of other emotion. Within the context of this thesis, instances of reported experiences reflecting these emotions will be classified as mental stress.

2.3.2 Physical stress

In this thesis, intense physical struggles such as heavy lifting or intense aerobic exercise are defined to fall under the category physical stress. It is important to note that this category not solely refers to the act of engaging in physical exercise, but rather the high degree of difficulty subjectively perceived by the individual doing the exercise. Evidently, this is dependent on the physical health of said individual. A metric strongly correlated to physical health is the heart rate recovery (HRR), which is the rate at which the heart rate drops after exercise and returns to its initial resting level. During this period, sympathetic activity withdraws, and parasympathetic activity increases to return to a restful state. Heart Rate Reserve (HRR) can be used as a marker of how intense the physical exercise was. As the intensity of an exercise increases, heart rate will also increase as well as the time for the heart rate to go back down to resting levels. Studies show that a high resting heart rate is associated with death from cardiovascular disease, and that abnormally low HRR is associated with sudden cardiac death. However, other studies also show that after cardiac rehabilitation on a number of patients, their heat rate after exercise significantly lowered and their overall resting heart rate also lowered after completing the program, indicating an improvement in parasympathetic function. [20] With this in mind, it becomes apparent that monitoring heart rate and how it improves over time becomes an important factor in assessing the physical health of an individual.

Incorporating heart rate monitoring into regular fitness assessments can provide crucial data for composing exercise programs to individual needs. This optimizes health outcomes, and potentially prevents adverse cardiovascular events such as hyperventilation or a stroke. By understanding and tracking these metrics, healthcare providers and individuals can work together to maintain and improve physical health through targeted interventions and lifestyle adjustments.

Chapter 3

Related works

Throughout the years, there have been several innovations to create technology with the purpose of stress management. This technology mainly seeks to quantify stress and classify emotional responses through the means of gathering biometrics such as heart rate data, like in the previous chapter. Various machine learning methods have been utilized to uncover underlying patterns in the data, enabling the automatic identification of potential indicators of stress or illness. In this chapter, a couple of existing technologies and studies will be described that are related to focus of this thesis.

3.1 Stress management & health monitoring

The first technology to discuss are the features provided by the Google Pixel Watch, a smartwatch developed by Google and Fitbit. By continuously monitoring the electrodermal activity (cEDA) sensor, skin temperature sensor, a multi-LED photoplethysmography (PPG) sensor to detect heart rate, a machine learning algorithm embedded in the watch can potentially identify indications of stress called body responses. These indications of stress can either be negative or positive, including arousal or excitement. When detecting a body response due to a sudden increase in electrodermal activity or drop in HRV, the smartwatch will trigger a notification and ask the user to reflect on the period in which the body response are detected by logging their mood. During the first month of wearing the smartwatch, the algorithm gathers data in order to determine personal baseline levels. From your personal physiology, the algorithm is trained to make accurate predictions in the future. After detecting a body response indicating negative stress, the smartwatch will also prompt the user to take part in a mindfulness session. If the user has bought Fitbit Premium, he will have access to several types of sessions including guided breathing and meditation. Another Premium feature is the Stress Management Score, which indicates the amount of stress responses in combination with logged moods and sleep patterns. In the Fitbit application, the user has a weekly overview of the body responses occurring per day and shows them in a small timeline along with the logged moods during those time periods.

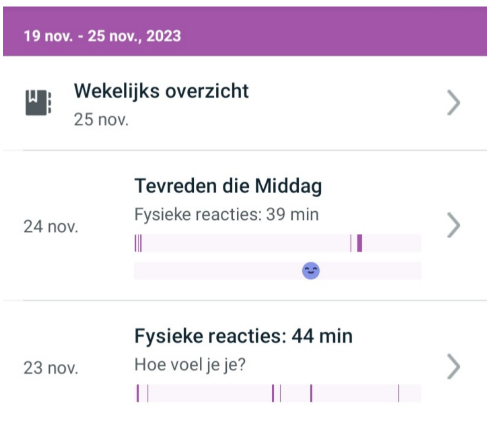


Figure 3.1: Stress responses timelines

Figure 3.2: Stress responses during the day

Employing a smartwatch for stress management is an innovative concept. While it has received much praise, the body response system it utilizes also has faced some criticism. According to some users, such as the ones that have expressed their concerns on the Pixel watch subreddit [29], the Pixel watch has often shown to make inaccurate predictions about how they were feeling. For instance, when users were feeling nothing in particular, the watch 'bothered' them with unnecessary notifications. In contrast, some users who perceive stress don't receive any notifications at all. During other instances, the smartwatch notifies the user to reflect on stressful moments that have occurred several minutes ago. As a result, some users can't recall exactly how they were feeling anymore. Some users turn off the notifications altogether because it bothers them too much. It's unclear from these reviews whether the Google algorithm is faulty or if users haven't logged their mood enough to adequately train the algorithm. From a blog post, Google has claimed to have successfully verified the smartwatch's ability to detect stress by performing a stress test by doing a surprise math test and simulating a dream job interview [39]. However, no research paper has been published that discloses information about the algorithms and machine learning model trained to detect body responses.

Another example of a commercial wearable device is the clinically validated Karios wristband developed by Biostrap. Just like the Pixel Watch 2, it uses an optical photoplethysmography (PPG) sensor to measure changes in blood volume. This sensor emits light onto the skin and detects the light absorption changes, which allows for PPG waveform analysis to detect events such as heartbeats. The wristband also has built-in accelerometer and gyroscope sensors to measure physical activity and analyse sleeping patterns, and also assists in removing motion artifacts in the PPG signal. Data is transmitted to a mobile app via Bluetooth, then processed in Biostrap's cloud servers using the Pulse Engine™, which filters out artifacts and extracts vital biometrics such as heart rate variability and respiratory rate. The wristband comes with the Vital Science app, which grants the user more insights into their biometrical data such as their stress data shown in the figures below.

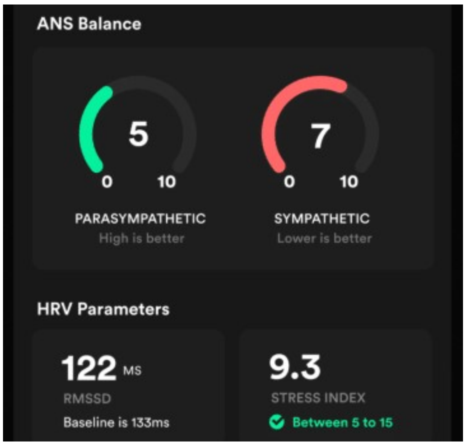


Figure 3.3: Vital Science ANS balance data [2]

Figure 3.4: Vital Science heart rate data [2]

Aside from detecting signs of stress, well-being and sleeping patterns, wristbands might also able to detect patterns related to cardiovascular disease. Gielen et al. (2020) presented a case study to assess patterns in physiological data such as resting heart rate, respiratory rate, oxygen saturation (SpO2), and arterial stiffness (AS) in individuals with severe acute respiratory syndrome (SARS-CoV-2) using the Biostrap wristband. Data was collected from 993 participants, resulting in a total of 2.5 million hours of biometrical data. The study presents two patients, one with no symptoms other than fatifue and one who experienced a loss of taste, smell, shortness of breath and decreased exercise capacity. In both patients, patterns of elevated heart rate and respiration rate were observed while arterial stiffness declined in comparison with healthy subjects. [15]

Another popular wearable device is the WHOOP performance optimization system, which is a wristband used by athletes to measure training performance, recovery, sleep and strain. Just like the Pixel Watch and the Biostrap wristband, the WHOOP wristband is equipped with a tri-axial accelerometer, optical sensor and an ambient temperature sensor. The measured data is sent through a cloud platform, and displayed in the WHOOP mobile application, showing a physiological profile to monitor adaptation to training, such as the stress monitoring screen shown in Figure 3.5. Emily A. Lundstrom et al. (2023) conducted a study using the WHOOP wristband to investigate links between energy deficiency (ED) and health parameters. This was done by analysing how the wearable data on heart rate variability (HRV), resting heart rate (RHR), exercise recovery, laboratory measures of metabolism (resting metabolic rate (RMR), total triiodothyronine (TT3)) and stress levels measured via the Recovery-Stress Questionnaire for Athletes (RESTQ) correlated in NCAA swimmers during heavy training. The results of this study showed no significant differences between sexes were found in any WHOOP or RESTQ variables. However, a negative correlation between heart rate variability (HRV) and sport-specific stress was observed, as well as total stress, indicating higher HRV correlated with lower stress levels across all participants. [22]

Figure 3.5: WHOOP stress monitor screen

3.2 Biometrics and machine learning

Imec (Interuniversitair Micro-Electronica Centrum) is a research center located in Leuven, Belgium, with a specialized focus on microelectronics, nanotechnology, and artificial intelligence. In 2018 they conducted a large-scale study called the Stress in the Work Environment (SWEET), which aimed to establish a link between physiological stress symptoms and self-reported stress in real-life. Data from 1002 participants was collected over a span of five consecutive days through various methods including psychological assessments, wearable sensors, ecological momentary assessments (EMAs) through a smartphone application, and stress tests. The wearable sensors used in this study were a chest patch used for measuring ECG signals and acceleration, and a wrist-worm device to measure skin temperature, skin conductance and acceleration called the Chillband. During the five days of measurement, the participants were notified through the smartphone application to assess their level of stress twelve times per day at random times, at least 30 minutes apart. During the first day of the experiment, individual stress responses were assessed to a known common stressor by using the Montreal Imaging Stress Task. This includes a 5-minute relaxation phase with soothing music and visuals, followed by a 5-minute segment where participants perform simple math tasks without time pressure or social influence. Next is a 5-minute stress-inducing phase where participants must complete math tasks within a time limit and under social scrutiny. Finally, there's another 5-minute relaxation phase with calming music and images. Contextual information collected throughout the five days included location, audio, movement and environmental sensors. 18 physiological features were investigated, including ECG and SC measurements, as well as accelerometer-based activity. These features were calculated over 5-minute windows with a 4-minute overlap to ensure accurate computation, particularly for HRV features like RMSSD. Later on, the features were used to train Random Forest machine learning models to predict stress levels based on physiological responses. The study revealed that strong statistical correlations exist between physiological responses, self-reported contextual factors, and behavior. [35]

In a study by Giannakakis, Marias, & Tsiknakis (2019) [14], heart rate parameters were investigated in order to reliably train machine learning models for stress detection. ECG signals were measured using a traditional lead sensor and RR-peaks had to be detected using peak detection and other preprocessing techniques. To examine the impact of stress, participants were subjected to various stress-inducing situations simulating common real-life scenarios. These stressors were organized into four phases, including social exposure, recalling stressful events, cognitive tasks, and viewing stressful videos, aiming to encompass different types of stressors. The study

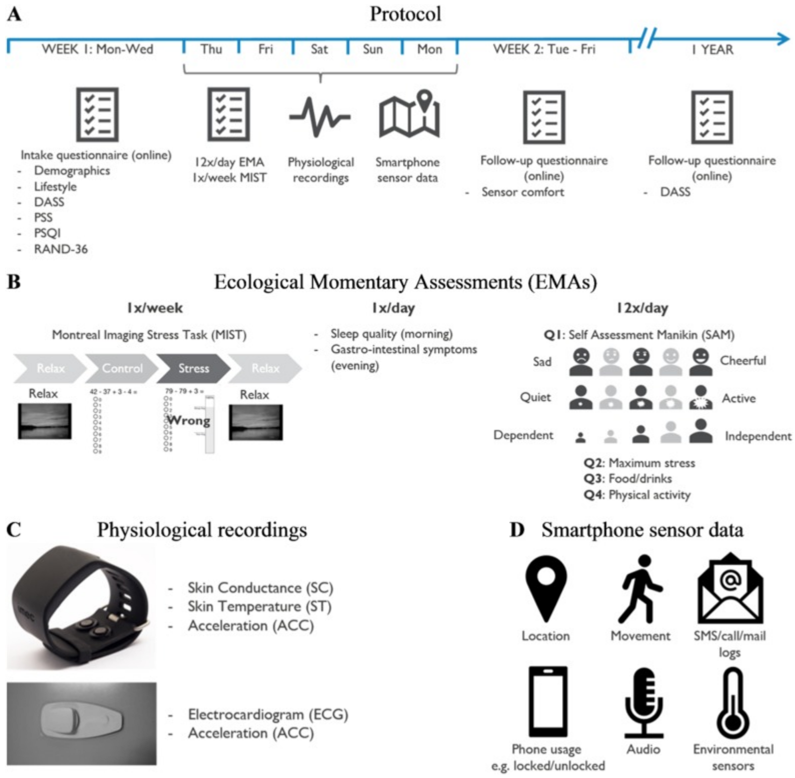


Figure 3.6: Imec SWEET study protocol [35]

involved 24 participants (7 women, 17 men) with an average age of 47.3 years. Data collection occurred during the second phase of a research project focused on monitoring cardio-metabolic risk, known as SRD'15. The calculated heart rate parameters form an $N \times M$ feature matrix X, in wich M represents the features and N the number of samples.

In order to normalize the data and remove individual baselines, this study viewed the problem of detecting stress as a ranking problem. The input consisted of the heart rate feature matrix X and a corresponding class vector Y. In order to transform this into a two-class classification problem (stress vs no stress), a pairwise transformation method was used. This method involves creating preference pairs of feature vectors and their associated labels based on temporal indices of stress and non-stress periods. For each pair of instances (i, j) where i and j are temporal indices representing the moments of sampling different time points, their stress labels are compared (0 and 1). If the stress label detected from the sample at time point i is higher than at time point j, the pair is labeled as a positive instance, indicating stress. If the stress label at time point i is lower than at time point j, the pair is labeled as a negative instance, indicating no stress. The resulting feature matrix X' and class vector Y' can then be used for a traditional classification problem. The study showed that after this pairwise transformation method, more accurate predictions were made by several machine learning models except Random Forest (RF), which showed a 5% decline in accuracy.

$$X' = \{X(t_i) - X(t_j)\} \qquad\qquad Y' = \text{sign}\{Y(t_i) - Y(t_j)\}$$

To select the most important features as shown in figure 3.8, the study used the minimum Redundancy Maximum Relevance (mRMR) selection algorithm, which selects a subset of features having the most correlation with the output class and the least correlation between themselves. After selecting and ordering the most important features, the misclassification error for each number top-ranked features was assessed to determine the optimal number of features to be retained as shown in figure 3.7. Using only ECG signals, the study showed a stress detection accuracy of 84% using Linear Discriminant Analysis (LDA) learning. This is quite good, since cardiac signals only contain limited information about stress.

Other studies have tackled the problem of stress detection using physiological data from multiple modalities. Schmidt et al. (2018) introduced the publicly available WESAD dataset designed for studying stress and affect detection using wearable technology. The dataset includes various sensor modalities such as blood volume pulse, electrocardiogram, electrodermal activity, electromyogram, respiration, body temperature, and three-axis acceleration. The dataset contains three classes: neutral, stress, and amusement, and incorporates self-reports from subjects through the use of questionnaires. The comprised physiological and motion data is derived from 15 graduate student subjects in a lab setting, captured by RespiBAN Professional chest strap device and the Empatica E4 wristband. Exclusion criteria during the study were pregnancy, heavy smoking, mental disorders, chronic and cardiovascular disease. The study protocol had the participants go through three different affective states: neutral, stress, and amusement, each followed by guided meditation in order to return participants to a neutral state. During participation, the participants refrained from caffeine and tobacco consumption and avoided physically intense exercise. In the amusement condition, the participants watched a series of funny video clips, 392 seconds in total . In the stress condition, the participants underwent the Trier Social Stress Test (TSST), which involves a five-minute speech in front of other people. After the amusement and stress conditions, the participants engaged in a seven-minute guided meditation to returning them to a neutral state through controlled breathing exercises. After the data collection, five machine learning models were evaluated: Decision Tree (DT), Random Forest (RF), AdaBoost DT (AB), Linear discriminant analysis (LDA) and k-nearest neighbours (kNN). The performance of these models was evaluated based on the use of different modalities and on binary classifaction (stress, no stress) and three-class classifcation (amusement, stress, no stress). The ensemble-based methods (Random Forest and AdaBoost) along with Linear discriminant analysis demonstrated similar performance in both the three-class and binary classification tasks. They achieved scores of up to 80% for the three-class problem and up to 93% for the binary task, depending on the input modalities. However, k-Nearest Neighbors (kNN) consistently had the lowest performance, with accuracies reaching at most 60% for the three-class problem and 78% for the binary task.

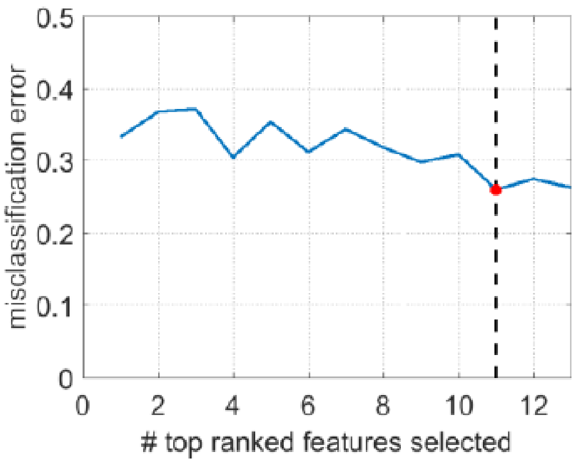


Figure 3.7: Misclassification error per selected amount of top features [14]

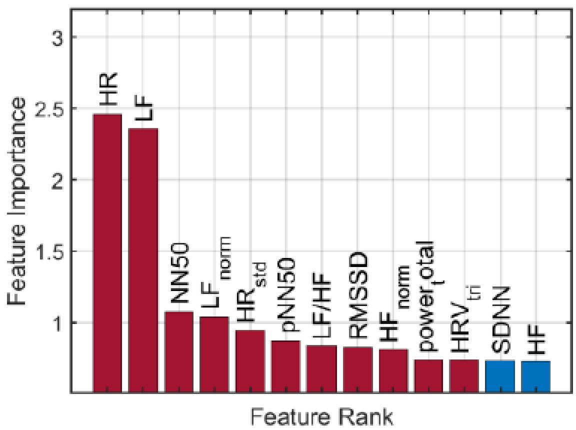


Figure 3.8: Features ranked by importance [14]

Although not all decisions made in this thesis were directly influenced by these studies and other similar studies that aim to detect stress through the measurements of biometrics, they have provided valuable insights into the appropriate processing and understanding of heart rate features before integrating them into machine learning models.

Chapter 4

Sensorics & data collection

4.1 Polar H10

4.1.1 Description

In order to obtain the heart rate of an individual an ECG monitor is needed. Typically, such measures are done in clinical environments using a hospital grade ECG monitor and attaching several electrode leads to the body that connect to the monitor. Obviously, this method is impractical during activities such as working or sporting due to its obstructive nature. Therefore, it is not suitable for continuously monitoring the heart rate. Even the Holter monitor, which is a portable ECG device can be considered too obstructive during physical activities. Having multiple leads attached to the body and a monitoring device hanging by the hip can be uncomfortable. Depending on the type of physical activity, one or multiple leads may rip off when the cables are accidentally being pulled. The solution is using a portable device that is non-obstructive and performs measurements that are as accurate, or at least come close to being as accurate as hospital-grade ECG monitors.

Figure 4.1: Polar H10 monitor

A device that suits these criteria is the Polar H10 heart monitor, which is a data processing unit connected to a flexible chest band equipped with a single-lead electrode. The sensor is predominantly used by athletes to measure their performance and by individuals who want to have more insight in their physical health. The chest band itself is called the Polar ProStrap, and is designed to reduce signal noise in order to measuring the heart rate accurately during physical activity. Studies have shown that the chest band is able to perform measurements as accurate as with a hospital grade ECG monitor [31] The chest band has a sampling rate of roughly 130hz and is able to provide a timestamp for each measured value in millivolts. The sensor is powered by a CR 2025 lithium coin battery and has a total lifetime of 400 hours.

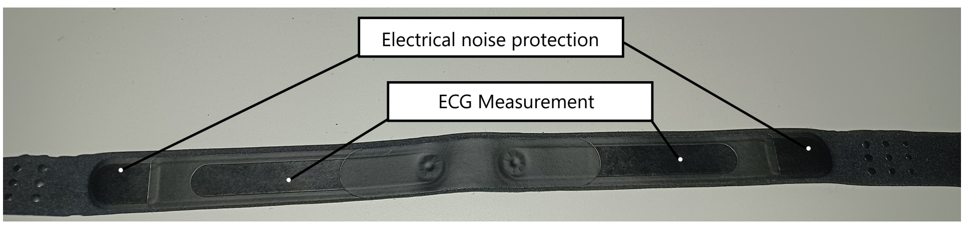


Figure 4.2: Polar Prostrap

The ECG signals measured by the Polar H10 are stored and processed internally in order to calculate the RR-intervals. The processing unit applies a real-time signal detection algorithm to accurately identify QRS complexes from the ECG signal with sub-millisecond precision. When compared with a real ECG signal, the Polar H10 has a very low error rate compared to wearable sensors such as the Garmin and Suntoo smartwatches [28] and will therefore be used as the tool to measure heart rate data in this thesis. Data from the sensor is acquired wireless via Bluetooth, with each manufactured sensor having its own unique identifier. The Polar Beat app, for example, makes use of this connection to let the user initiate a physical activity session. The characteristics of the session can be stored into the eternal memory of the sensor itself, which can later be uploaded to a personal dashboard if the individual is logged in into their Polar account. The Polar H10 is also equipped with a tri-axis accelerometer which is used for movement detection purposes such as counting steps, but this feature will not be explored in this thesis.

4.1.2 Data processing

As mentioned before, the data calculated by the Polar H10 can be obtained by opening a Bluetooth connection with the processing unit. An input stream is used to obtain the data in real time, which will then be buffered and saved into a file. There exist several applications on the Play Store that provide functionality to detect a nearby Polar device and open a connection with it. One of these applications is called "Polar Sensor Logger", and has the capability to connect to a sensor via the Polar SDK and store RR-interval data, HR and HRV data as well as the raw ECG data into text files. With each value, the timestamp with millisecond precision is also stored. This functionality makes it unnecessary to manually calculate the RR-peaks from the ECG signal. As mentioned in the second chapter, stressful events lead to an increase in heart rate and a decrease in HRV according to literature. To validate this assertion and analyse this phenomenon, several recordings were conducted throughout several days. These recordings are called "perceived stress logs" and contain logs about stressful events which have occurred throughout the day, which include both physically and mentally stressful events. The perceived stress logs are paired with the heart rate recordings performed with the Polar Sensor Logger application in the form of text files. By making use of Python notebooks such as Jupyter Notebook and Google Colab, these files are be analysed and processed step by step. The primary libraries utilized for data processing are as follows:

- **pandas** is a library that converts datasets of various formats into a DataFrame. This object contains various functions for manipulating rows and columns, which makes processing the data easier.
- **matplotlib** is used to plot data into various types of graphs. In this analysis, it will be used to create box plots and line graphs.
- **HRVanalysis** is a library created by the Aura Health Project [5] for seizure detection, and provides various functions for performing HRV analysis such as outlier and ectopic beat filtering.
- **pyHRV** is another library for calculating HRV which addresses the shortcomings of the previous library such as calculating the pNN25 time domain feature.
- **heartpy** is a HRV library that provides a public method for selecting a bandwidth of an RR-interval signal. The resulting signal is used for calculating the breathing rate based on the RR-intervals.

The type of HRV that is focussed on during the initial analysis is the RMSSD, as it is the most common indicator for parasympathetic activity. Even though applications provide pre-calculated heart rate and HRV, all HRV features are calculated from the RR-intervals in this analysis. The first step in calculating the HRV, is to first and foremost clean the signal by removing and interpolating the outliers. Next is the removal of ectopic beats, which removes RR-intervals that don't stem from parasympathetic activity on the pacemaker of the heart known as the sinus node. Before and after the filtering of ectopic beats, any potential NaN values resulting from the prior calculations are removed and interpolated. The resulting list of interpolated NN-intervals will be used further on. This list will remain the same length as the original series of RR-intervals, but the timestamps given by the sensor are still associated with their original RR-intervals.

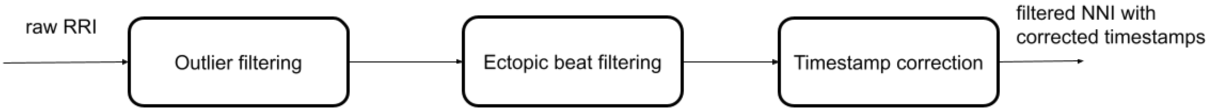


Figure 4.3: Signal cleanup pipeline

Next is the calculation of the heart rate and HRV using the sliding window approach. This method iterates over the list of NN-intervals and buffers them in a `deque`. The window sum controls how many elements are allowed to be stored within the queue. Because we want to calculate the heart rate and HRV based on one minute of recordings, the maximal sum is 1000×60 milliseconds. When the sum of buffered RRI exceeds one minute, the front of the queue is popped until the sum is less or equal than one minute. After this step, the heart rate and HRV are calculated from the NN-intervals in the buffer. In the next iteration, a new interval is added to the buffer. This process repeats itself until there are no NN-intervals left. The psuedocode snippet below shows how an NN-interval is inserted into the buffer. The `sum (buffer) > windowsum  1000` check in the `if`-statement prevents calculations when the amount of intervals in the buffer is not sufficient. The result is a list of heart rates and HRV each with their respective timestamp.

```
buffer = collections.deque()

def buffer_rri(timestamp, nni, windowsum =
    1000*60):
     def add_to_buffer(element):
          buffer.append(element)
          while sum(buffer) > windowsum:
               buffer.popleft()

     add_to_buffer(nni)

     if sum(buffer) <= windowsum and
        sum(buffer) > windowsum - 1000:

          //CALCULATE FEATURES FROM BUFFER
          return features
```

Code Fragment 4.1: Function to calculate heart rate features through the usage of an RR-interval buffer

4.1.3 Data recording

It is known that low HRV is associated with stress and anxiety, but what exactly is 'low'? "Twenty-Four Hour Time Domain Heart Rate Variability and Heart Rate: Relations to Age and Gender Over Nine Decades" [43] is one of the studies that has tried to discover the effects of age and gender on HRV, and has found a positive correlations between HRV and gender by monitoring 179 subjects of various age groups with a Holter system. For subjects between the age of 20 and 29, the study has conducted that normal RMSSD-calculated HRV has a normal range of 43 ± 19 ms and a heart rate range of 79 ± 10. While these findings are not necessarily incorrect, they do not represent the general population. Athletes, for example, quite literally train their heart muscle by doing exercise. As a result, the heart is able to pump more blood to the body per beat to provide oxygen to the muscles, meaning that the athlete's heart beats fewer times per minute compared to a non-athlete. Therefore, it is important to take individual differences into account. In this thesis, the notion of 'low HRV' is based on the statistical analysis of the individual's HRV values throughout daily life. In order to perform individual analysis on heart rate data during physically and mentally stressful events, it is required to possess some kind of baseline data for comparison. For this reason, several data recordings were conducted while working on assignments for courses at home. With the RR-interval samples obtained from these recordings, a list of heart rate and HRV values can be calculated using the sliding window method as described above. Using the `numpy` library, statistics can be derived from the lists including the mean and quartiles. This information can be visualized in a boxplot using `matplotlib`. Figure 4.4 shows a boxplot of the calculated HRV. The interquartile range shows that most of the recorded values lie between 50.61 and 82.36 milliseconds. The high HRV values resulting from the effects of one-minute calculations form the outliers on the high end. 'Low HRV' can be defined as any value falling beneath the first quartile. This is a deliberate choice, because mild stressors do not necessarily push the HRV directly to lower outlier values. Figure 4.5 shows a boxplot of the calculated heart rate. The interquartile range shows that most of the recorded values lie between 58.0 and 69.0 beats per minute. Values commonly associated with physical exercise are considered as less frequent and outliers, which is to be expected since heart rate seldom reaches those levels while working at home.

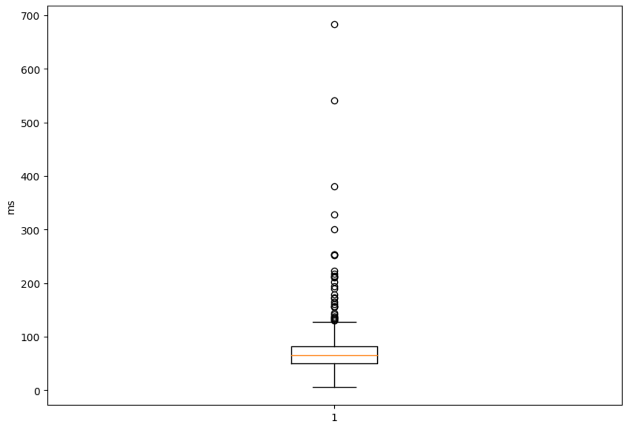


Figure 4.4: Sampled HRV boxplot

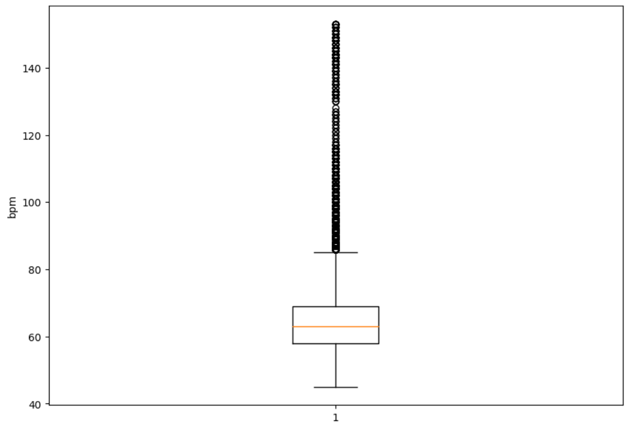


Figure 4.5: Sampled Heart rate boxplot

After having calculated these statistics, more recordings were conducted to analyse how physical exercise influences heart rate and HRV. First, two minutes of intense exercise were performed to validate the decrease of HRV during exercise and gradual increase during recovery after exercise. Figure 4.7 illustrates the duration of physical exercise, delineated by the zone between the yellow lines, and period of active rest between purple lines. As expected, heart rate goes up during exercise and goes back down when exercise is ceased, returning to baseline levels which are indicated by the broken lines representing the interquartile range of the sampled heart rate. While the exercise only lasted two minutes, it took more than 20 minutes for the heart rate to fully stabilize. Figure 4.6 shows how the HRV fluctuates during the period of recording. Before initiating the exercise, there are some drops and peaks in HRV. These sudden drops can be related to digestion and other uncontrolled factors. When starting the exercise, the variance increases and drops at the $\approx$ 1:30 minute mark. After ceasing the exercise, the HRV drops significanty under the first quartile. During the recovery period the HRV follows a general rising trend. There are intermittent peaks, albeit remaining below baseline. The rising trend in HRV corresponds to the literature describing parasympathetic reactivation.

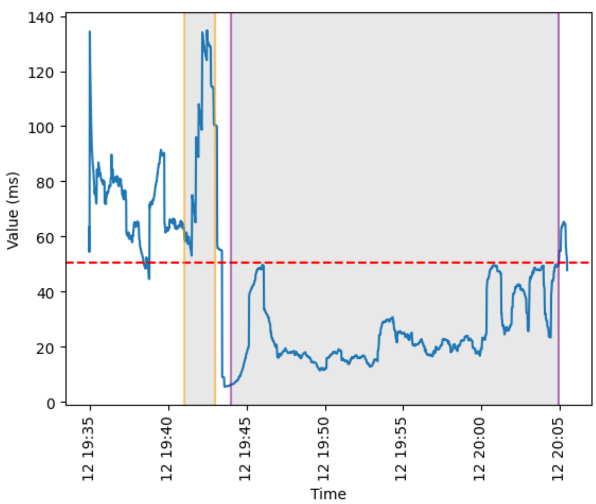


Figure 4.6: HRV during exercise compared to baseline

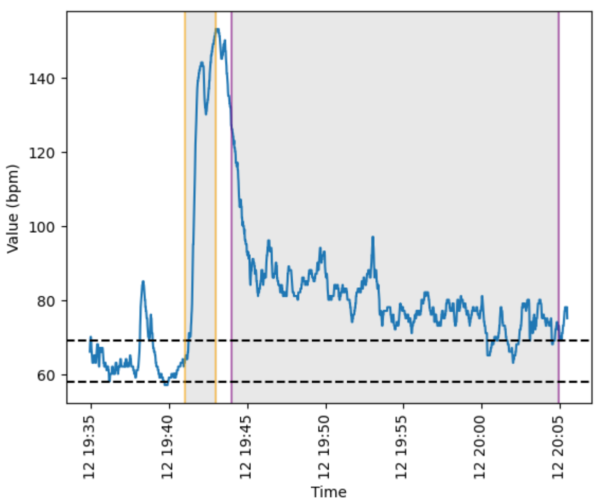


Figure 4.7: Heart rate during exercise compared to resting range

Since the length of this recording is relatively short, it is interesting to see how the heart rate and HRV fluctuate during an extended period of time of exercise. The chosen recorded session has a duration of approximately one hour, starting at **21:04**. First, five sets of barbell squats were performed, starting with a warmup set and gradually increasing weight during the subsequent sets up to heavy amounts. Between each set, a pause of a couple of minutes was held to prepare for the next set. After squatting, a rest period of eight minutes was held before performing three sets of pull-ups. At **21:45**, the training session came to an end. The recording ceased at **22:12**. Looking at figure 4.8, it becomes clear that the peaks and dips in heart rate correspond to the pauses between each set. During the rest period, there is a larger decrease in heart rate. Naturally, the mean of the sample lies well above the resting heart rate range.

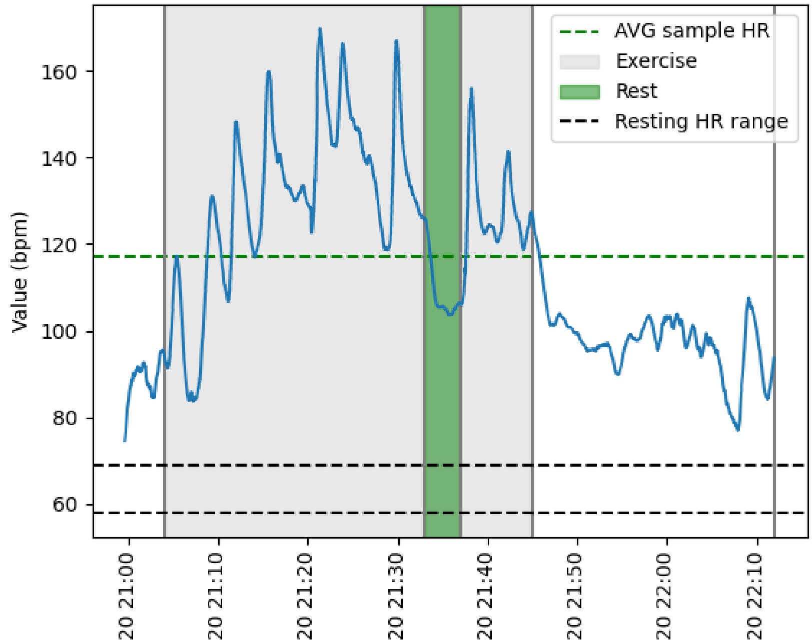


Figure 4.8: Extended Heart rate recording

Looking at the HRV values in figure 4.9, it becomes clear that the majority falls under baseline. After the resting period, HRV was restored to baseline levels. It is notable that before the resting period, there are instances in which HRV peaks above baseline which may be attributed to the resting periods in between sets. After the exercise, the HRV has some dips during its general rising trend, although no notable stressful events were logged during this period. These fluctuations can again be attributed to parasympathetic stimuli by the autonomic nervous system, related to other bodily functions besides stress.

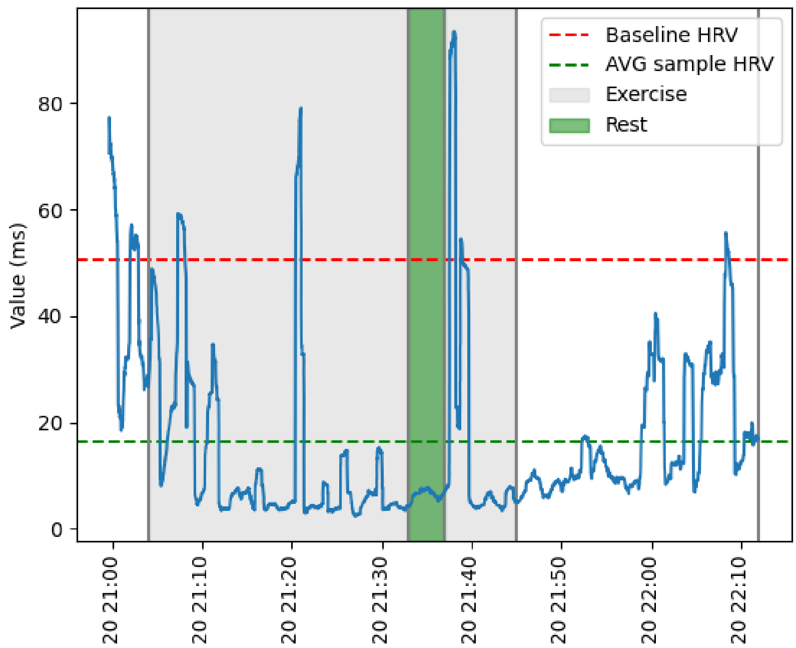


Figure 4.9: Extended HRV recording

Next is performing analysis on a recording during a mentally stressful event. The chosen recorded event to analyze, is being under deadline pressure. During the time of the event, there were unresolved issues with a project combined with tension among group partners. This led to the experience of negative emotions, predominantly frustration. Looking at figure 4.10, it becomes apparent that the spikes in heart rate are contributed by these feelings of frustration, resulting in an average heart rate above the resting heart rate range.

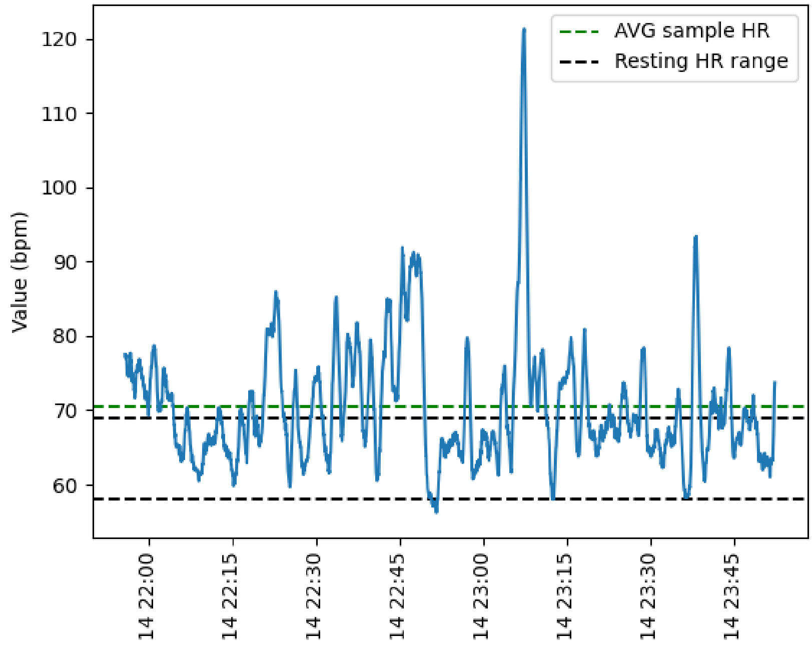


Figure 4.10: Extended HRV recording

The heart rate recordings in figure 4.12 show the same type of fluctuations. The line chart that when the heart rate peaks, HRV tends to drop. Examples are the large spikes between **22:15 - 22:30**, **23:00 - 23:15**, and **23:30 - 23:45**. The drops in HRV don't last however, in contrast to physical exercise where HRV stays low during an extended period of time. During the majority of instances, HRV is well above baseline. This is somewhat consistent with literature that emotional regulation leads to an increased vagally mediated HRV [21].

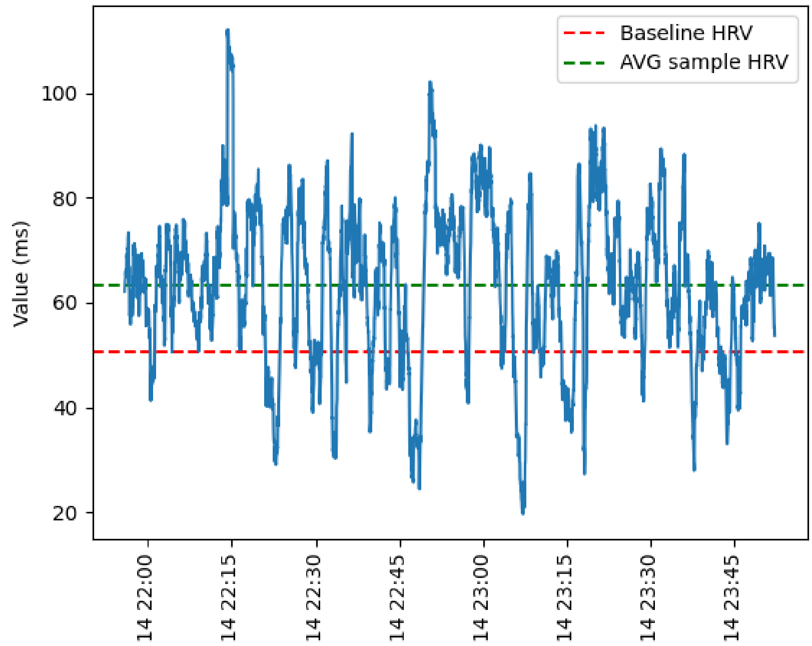


Figure 4.11: Extended HRV recording

Intuitively this makes sense. During physical exercise, the body is more or less “in need" of sympathetic dominance in order to pump blood to the muscles and promote alertness, whereas this isn’t the case when experiencing negative emotions during a non physically active state. While heart rate and HRV can give some insight into the health of an individual, it is important to note that solely looking whether HRV has dropped below baseline or not is insufficient to make predictions about stress. By incorporating machine learning techniques, a more reliable stress detection method can be obtained. How this works exactly will be explained in later sections.

4.2 Google Pixel watch 2

4.2.1 Description

The second wearable that seemed valuable to experiment with was the Pixel watch 2 developed by Google and Fitbit. Its functionalities allow the user to get information about physical health including minutes in active zones, sleep patterns, calories burned and the amount of steps taken. Furthermore, the watch possesses a module to detect and signify signs of stress detected throughout the day. Sensors attached to the watch include a multi-path optical heart rate sensor to calculate heart rate features, a 3-axis accelerometer to detect movement, an electrical sensor to measure skin conductance (cEDA) to assess emotional triggers and a skin temperature sensor. [16] On the smartphone, the user has access to the sensor readings by using the Fitbit application, which visualizes all the acquired data and derived conclusions. The smartwatch can be synced with user’s the phone via a Bluetooth connection to receive the newest data and smartwatch notifications. Data acquired by the smartwatch is stored in the cloud, and can be downloaded by means of exportation in .json and .csv files

Figure 4.12: Google Pixel watch 2 [13]

4.2.2 Sleep analysis

Since the smartwatch has the capability to analyse sleep patterns, it can be interesting to see how stress can affect sleep. In earlier sections it was made clear how HRV changes during daily activities, but these dynamics differ during sleep. That is because, during different stages of sleep, the autonomic nervous system switches from parasympathetic to sympathetic predominance. The sleep stages are divided in three categories: wakefulness, NREM (non-rapid eye movement) sleep which includes light and deep sleep, and REM (rapid eye movement) sleep. Literature shows that a transition from light sleep to deep sleep is accompanied by an increase in parasympathetic modulation. In contrast, a transition from NREM to REM sleep has shown to cause a decrease in HRV due to an increase in cortisol release. Research indicates that REM sleep occurring towards the end of the night is paired with increased sympathetic modulation in contrast to REM sleep observed earlier in the night. [26] This is because cortisol levels are in tune with the cardiac rhythm; cortisol levels typically begin to increase in the early morning, somewhere around 3 am, and reach their peak within the first hour after waking. This phenomenon contributes to feeling awake and energized in the morning, and is known as the cortisol awakening response. Increased general cortisol levels due to stress can excessively spike cortisol levels during the night, causing you to wake up in later stages of the night [30]. However, elevated cortisol levels during the night can also be contributed to a medical condition or sleep disorders. Insomnia, for example, is a sleep disorder which causes a person to fall and stay asleep or wake up too early, leading to mood alterations and fatigue throughout the day. Studies indicate that individuals with insomnia experience heightened sympathetic modulation both while awake and during the night, which may contribute to an elevated risk of cardiovascular diseases. [40] With this information in mind, it becomes interesting to see to what extent the data recorded by the watch can reflect these phenomena. From wearing the watch during the night, the smartwatch is able to detect the different sleep stages as well as the RMSSD, HF and LF HRV features. It is noted that the smartwatch only provides the HRV features in intervals of five minutes. Furthermore, the smartwatch only seems to be able to perform calculations during the night as heart rate and HRV measurements weren't accessible in the export files. Figure 4.13 shows a HRV recording during the night of 6 december. During the early stage of the night, HRV seems reasonably low, but according to individual experience it takes a while to completely fall asleep. During the middle of the night there is a significant drop in HRV, possibly indicating the spike in cortisol. REM sleep, however, seems to be accompanied with higher HRV throughout the graph while literature suggests the opposite. In the morning, the HRV has a few dips which can be possibly be attributed to being woken up in the morning. Other recordings show a similar rising HRV trend during some of the REM regions, except for the dip during the middle of the night which all recordings share, but overall the HRV compared to the regions varies greatly. Due to the lack of medical expertise to form grounded conclusions and the ambiguity of these readings, sleep analysis will not be discussed any further.

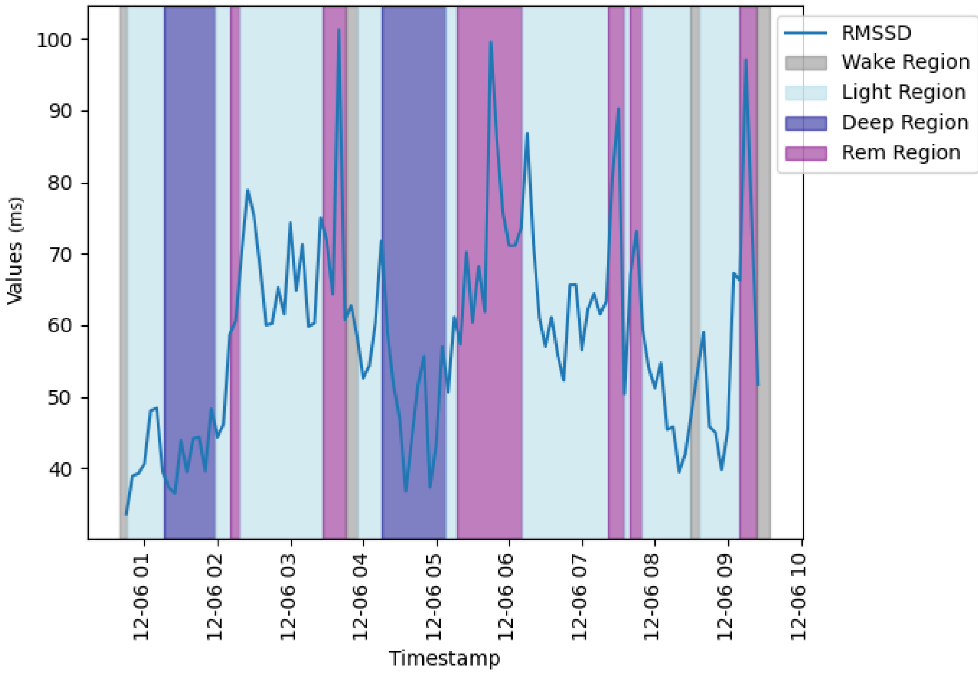


Figure 4.13: Nighttime HRV recording 6 december

4.2.3 EDA-levels

In Chapter 2, it was briefly mentioned how skin conductivity can serve as an additional metric to indicate stressful events. In this section, the events written in the perceived stress logs will be compared with the recorded EDA data from the smartwatch. Exports from the smartwatch provides EDA-level values in intervals of one minute. However, EDA-levels are not recorded during nighttime. To get a notion of what levels are 'high' or 'low' for the individual, it is useful to know what the average is throughout the days. This is visualized in the boxplot in figure 4.14, which shows most of the values lie between 2.33 volts and 4.9 volts. The average level is 3.8 volts, and the mean 3.65. The first plot in figure 4.15 shows the EDA-levels throughout the entire day. Two notable events were logged, which are indicated by the grey zones in the plot: going for a walk in the cold after feeling weary from studying around **15:55**, and weightlifting around **20:45**. In the first event, the rising trend in EDA-levels could be explained by the increasing weariness during studying inside, prompting a break which causes the drop in EDA-levels. Surprisingly all EDA-level values are above the mean marked in red, which might indicate a stressful day. In the second event, it is obvious that heavy physical exercises causes sweating which made the EDA-level spike. The second plot in figure 4.16 shows a recording before and after going to bed at **23:15**. The stressed logs showed sleeping issues. However, all other recordings show a rise in EDA-levels when going into bed, possibly to the rise in temperature from wearing the watch under blankets. Therefore, it is unclear to what extent the perceived stress has influenced the EDA-levels. The last plot in figure shows how EDA-levels rise from the moment stress during studying was reported at **17:15**. From these recordings alone, it is unclear how stress affects the EDA-levels. Studies that show positive correlations between EDA-levels and emotions are usually peformed in a controlled environment in which the influence of artifacts such as environmental temperature is minimized, in contrast to daily recordings with the smartwatch.

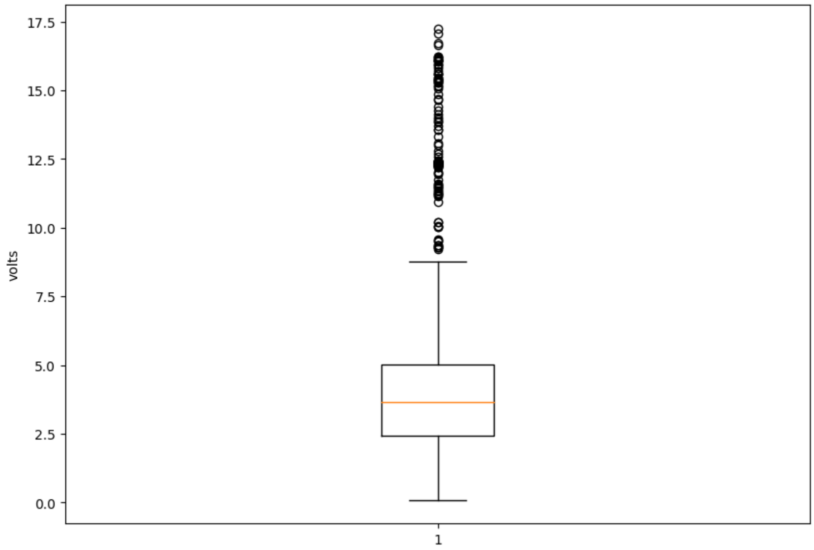


Figure 4.14: EDA-levels boxplot

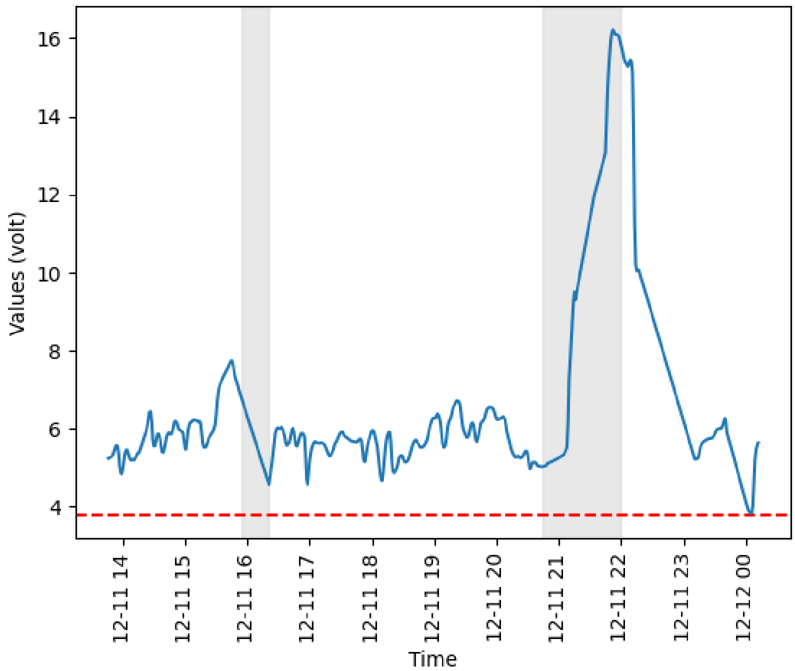


Figure 4.15: Daily EDA levels of 11 december compared with mean

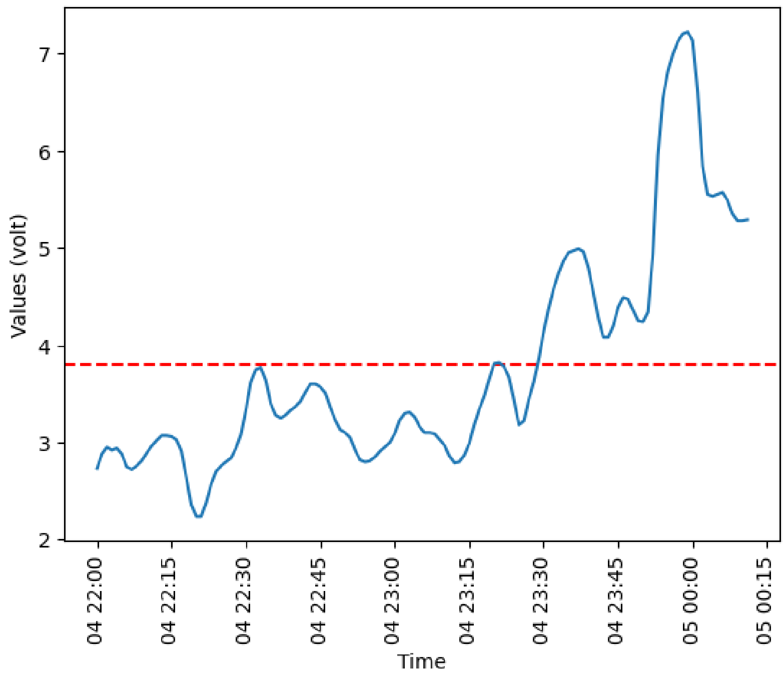


Figure 4.16: EDA levels of 4 december before sleep compared with mean

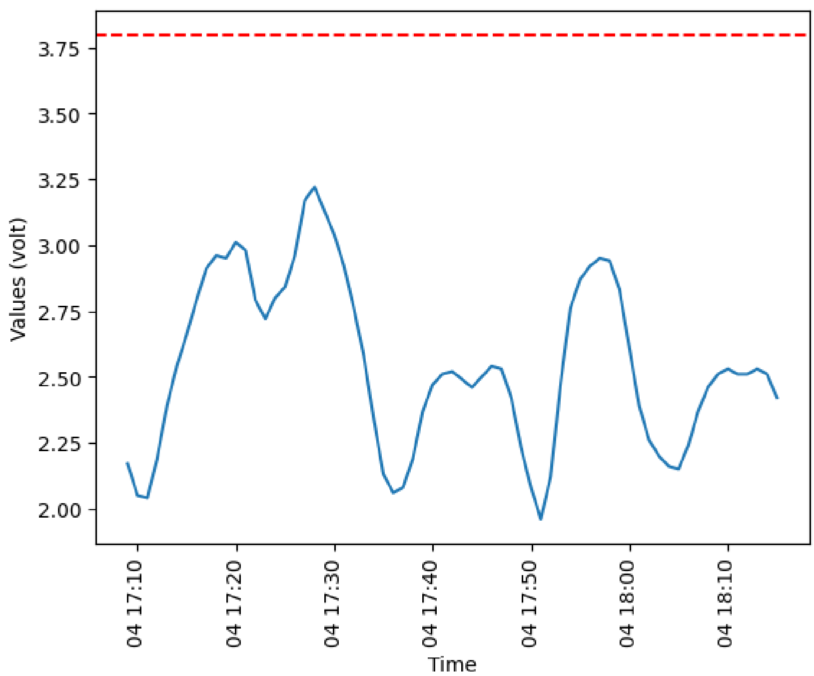


Figure 4.17: EDA levels 4 december compared with mean

Chapter 5

Stress management system

Looking back at the problem statement, it is desirable to efficiently monitor the well-being of an individual in an efficient and non-obstructive manner. Based on the background information in the previous chapters of this thesis, a stress management prototype system was developed and tested as a solution. The goal of this system is to effectively measure and visualize a variety of mental and physical health parameters, including physiological data and subjective feedback. Through this information, the user can be conscious about his overall well-being, identify patterns, and make informed decisions regarding his health. In this chapter, the system and its technical aspects will be discussed thoroughly along with the design choices taken during development.

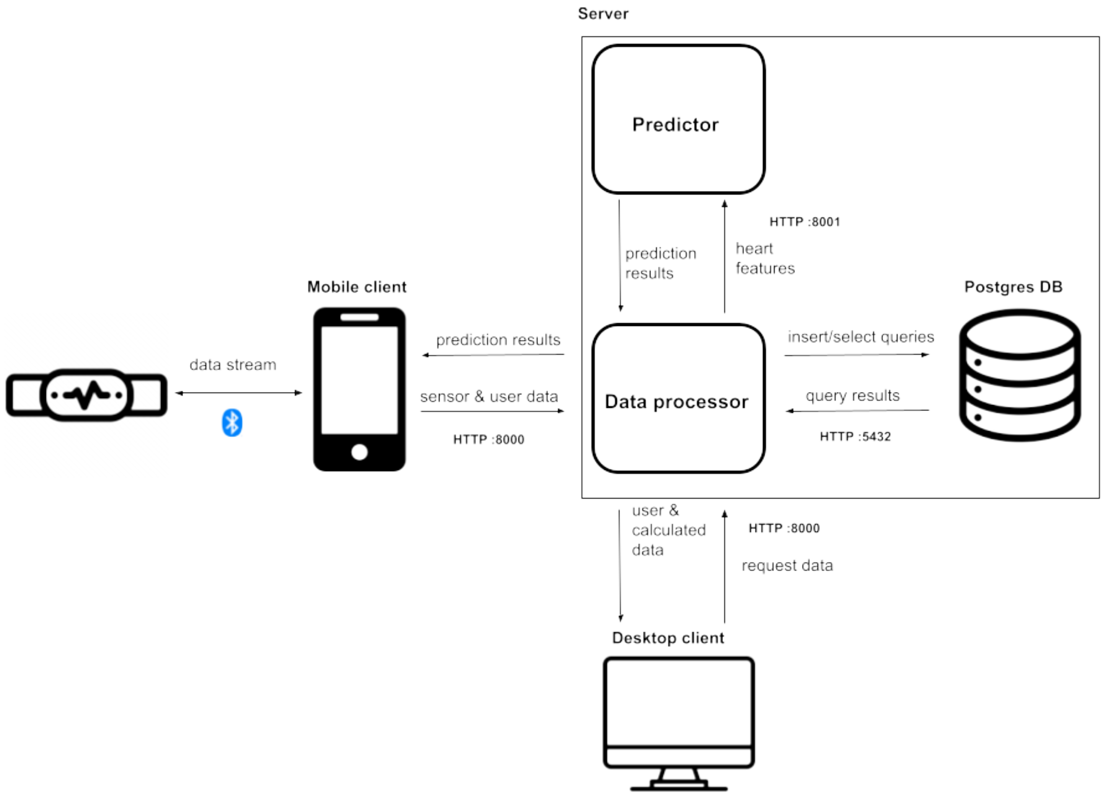


Figure 5.1: System architecture

Figure 5.1 above shows an overview of the developed system's architecture, consisting of four major components. A mobile application is used to connect to the heart rate sensor via a Bluetooth connection, and to log moods and activities. The data acquired by the mobile application is passed to the data processor via a HTTP connection on the local network. The data processor handles the transformation of sensor data into different features, and handles the logged subjective feedback. This information is stored in a PostgreSQL database server running on the same machine. To make predictions about stress based on heart rate features, data is passed from the data processor to the stress predictor, which passes its predictions back to the data processor. The data processor in turn passes the prediction to the mobile application to notify the user, and stores it in the database. The web application asks the data processor to retrieve the stored data in order to visualize it in an intelligible dashboard. Unfortunately for the smartwatch, no direct method was found to directly access the EDA-values via a Bluetooth stream. Therefore, no functionality is written for the smartwatch except for the inserting values from the .json export into the database. As an alternative for instances in which the heart rate sensor was worn but no EDA-values were recorded, these values will be mocked via a script instead.

5.1 Mobile application

The mobile application is developed in Kotlin using the Polar SDK, created for developing Android and iOS applications for Polar sensors. It is build on top of a starter project provided in the Polar Github repository, which contains functions to communicate with the H10 sensor. When connecting to the Polar H10 sensor, the SDK is used to stream data from the sensor to the application and to manage internal storage on the sensor itself. [12] The SDK uses Java ReactiveX. This is a library for composing asynchronous and event-based programs by subscribing to Observable sequences. It is particularly useful for mobile applications for listening to streams from API's while maintaining UI functionality. Reactive programming offers a functional programming alternative to reading data streams, making it more readable and easy to work with compared to its imperative programming counterpart. The Observable interface differs from the Iterable interface, in the sense that the producer (Observable) 'pushes' values to the consumer whenever values are available. This is in contrast to 'pulling' from the producer in Iterable, in which the main thread blocks until the values arrive. [3] In the case of the Polar H10 sensor, the program subscribes to the Bluetooth stream opened between the phone and the sensor and processes sample data whenever its available, as shown in the code fragment below. With this data, UI textboxes showing the current heart are updated, and RR-intervals are directly passed to the data processor. An example is shown in Code Fragment 5.1 below:

```
hrDisposable = api.startHrStreaming(deviceId).subscribe(
                    { hrData: PolarHrData ->
                        for (sample in hrData.samples) {
                            //process sensor data
                        }
                    },
                )
```

Code Fragment 5.1: Listening to the Bluetooth sensor stream using Reactive programming

Since the application is only meant to record sensor data and log activities, its usage is pretty straightforward. Nevertheless, the application needs to be intuitive and needs to fulfil the requirements needed for data acquisition. The user starts by connecting to the worn sensor, having provided the corresponding device identifier which is shown on the back of the sensor. If the sensor is connected, the user can initiate an activity or log its mood. When finishing an activity, the application will ask the user to log its mood in order to train the machine learning algorithm in the back-end on the heart rate data from the activity, labeled by the logged mood. While connected to the sensor and the API, the user can receive a message from the API in case a stressful event is detected. When this is the case, the application will generate a push notification to inform the user of the stress detection. Having clicked on the notification, the application will again ask the user to log his mood in order to improve the accuracy of the machine learning model. This flow is shown in the activity diagram in Figure 5.2 below.

The mobile application needs to do several things simultaneously: listen to detected stress messages from the data processor API, send RR-intervals to the data processor API, log a mood or an activity and send it to the API, and listen to UI events. Such operations happening simultaneously may cause indefinite delays on networking operations and the cessation of responsiveness to user actions on the UI. For that reason, the application needs to make use of multiple threads in order to prevent it from crashing. Figure 5.3 shows how a diagram of how this works in the application. The main thread, or the UI thread, is responsible for listening to the sensor Bluetooth stream and updates the UI accordingly to show the user's current heart rate, the activity timer and other information provided by the sensor. Of course, the main thread needs to listen to UI events such as clicking on a button

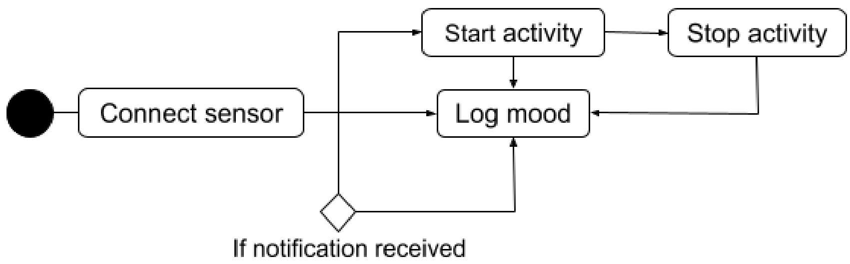


Figure 5.2: Mobile activity diagram

to change state of the application. In order to communicate data between two threads, a producer-consumer paradigm is applied. While the main thread gathers RR-interval data from the sensor, it sends it to a LIFO-queue. This queue is popped by another thread whenever values are present in the queue, and sends them to the data processor API. To ensure the UI doesn't crash under any circumstance, the main thread spawns another thread to handle sending an activity or mood, and kills it directly after.

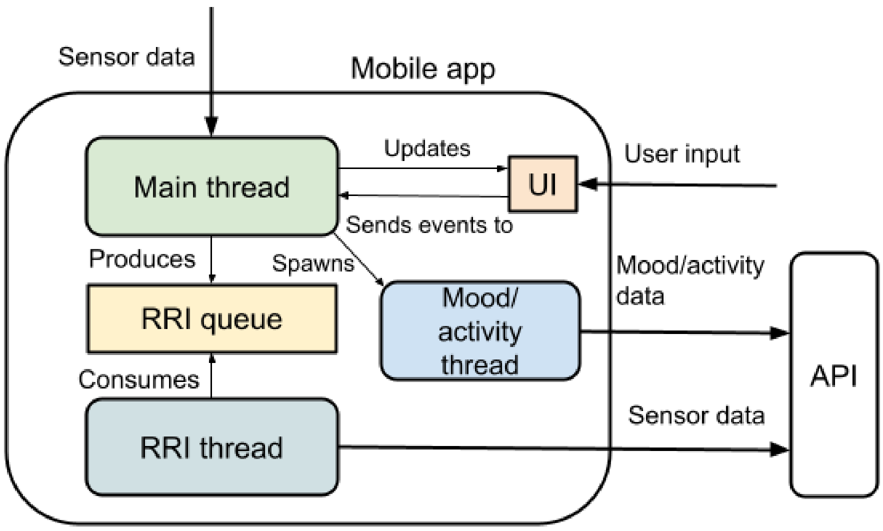


Figure 5.3: Inner workings of the mobile app

The application consists of four windows in total. When starting up the application, the user finds himself at the start screen in Figure 5.4. Pressing on the second button lands the user to the main page in figure 5.5, in which the application will find a Polar device via Bluetooth. When the device is connected, the heart rate of the user will be shown. As explained by the activity diagram, the user has three options from here. With the first button, the user navigates to the screen in Figure 5.7. Here, the user can choose an activity type and enter a description. When the user submitted the form, the timer will start in the main screen. With the same button, the user can stop the activity timer and will be sent to the screen on Figure 5.6. In the meanwhile, the activity with its start time, end time and description will be sent to the API and stored in the database. This screen is also opened using the second button in the main screen. In the mood screen, the user can select his mood and write a description. Having pressed the "log mood" button, the user will be redirected back to the main screen and the mood along with the description will be sent to the API and stored. The mood type "exhausted" corresponds with the class "physical stress", whereas the mood type "stressed" corresponds with the class "stress" which stands for mental stress. The rest of the mood types correspond to the class "baseline".

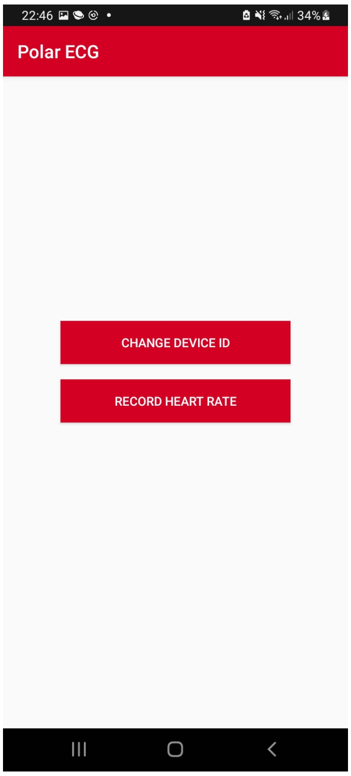


Figure 5.4: Start screen

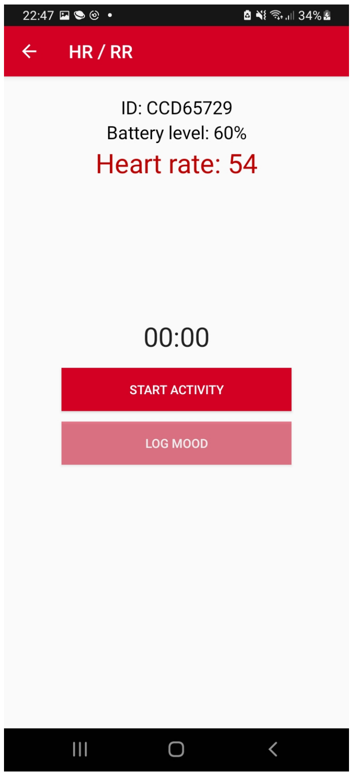


Figure 5.5: Main screen

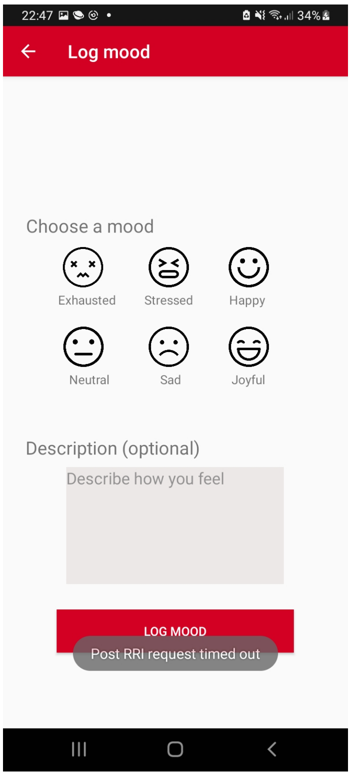


Figure 5.6: Mood selection screen

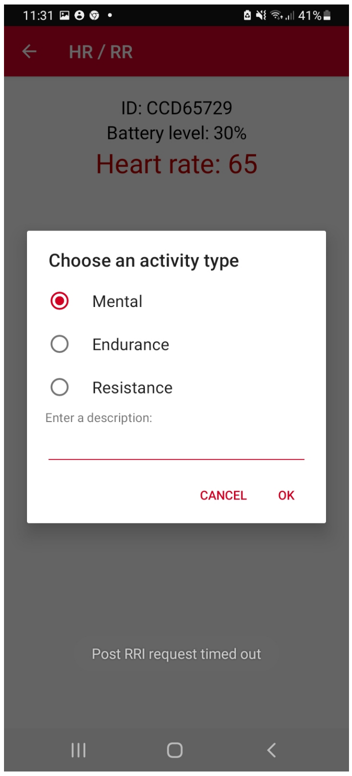


Figure 5.7: Activity selection screen

5.2 Data processor API

The data processor API serves as the backbone of the system. It is mainly responsible for processing the incoming data and converting the recorded RR-intervals into heart rate features, pass them to the stress predictor and store them into the database for visualization purposes. The back-end is entirely programmed in Python. To handle HTTP requests to the server from the web and mobile applications, it makes use of the FastAPI library. In contrast to other REST API frameworks such as Java Spring boot or C# ASP.NET, FastAPI is lightweight and is set up very easily unlike the former. This makes the library ideal for fast prototyping and simpler use cases. Integrating the functionality of processing data into the existing mobile application would introduce additional complexity. Additionally, lacking familiarity with machine learning libraries in Java, the API solution in Python was chosen. This choice allows for simpler algorithm implementation and load balancing, but necessitates a network connection as a downside. The table below shows a summary of the HTTP endpoints provided by the API and their respective descriptions, which explains where the data is coming from.

Table 5.1: Description of API endpoints

Endpoint	Description
@app.post("/postrri")	Expects a single RR-interval value and buffers it for the data processor.
@app.post("/postmood")	Expects a logged mood with its description for storage, and calls the predictor to train the machine learning model based on the mood type.
@app.post("/postactivity")	Handles a logged activity and stores its duration and type in the database.
@app.post("/")	Clears the list of buffered RR-intervals for data processing in case the mobile application ceases recording.
@app.get("/getstress")	Communicates to the mobile clients whether stress was detected or not. This endpoint is called every five seconds to avoid unnecessary requests.
@app.get("/getmood")	Used to retrieve the logged moods and their information in a specific time range.
@app.get("/getactivity")	Used to retrieve the information of logged activities in a specific time range.
@app.get("/getrespiratoryrate")	Used to retrieve the calculated respiratory rates in a specific time range.
@app.get("/getHRV")	Used to retrieve the calculated HRV values in a specific time range.
@app.get("/getheartrate")	Used to retrieve the calculated heart rate values in a specific time range.
@app.get("/geteda")	Used to retrieve the measured EDA levels in a specific time range.
@app.get("/getstressresponse")	Used to retrieve the detected instances of stress by the predictor in a specific time range.
@app.get("/gethraverages")	Used to retrieve the average heart rates from the preceding x days/weeks/months based on the current date.
@app.get("/getHRVaverages")	Used to retrieve the average HRV values from the preceding x days/weeks/months based on the current date.
@app.get("/getmoodcounts")	Used to retrieve the amount of logged moods and their types from the preceding x days/weeks/months based on the current date.
@app.get("/getstressresponsecounts")	Used to retrieve the amount of detected instances of stress from the preceding x days/weeks/months based on the current date.
@app.get("/getphysicalactivityminutecounts")	Used to retrieve the amount of minutes spent in physical activity from the preceding x days/weeks/months based on the current date.

@app.get("/getaveragesperactivitytype")	Used to retrieve the averages of physiological measurements for a specific activity type with the purpose of comparing it to other instances of said type.

After sending an RR-interval to the API, the same function from Code Fragment 5.1 is used to buffer the RR-interval. Each minute, the heart rate features are calculated from the RR-intervals. After calculating the features, heart rate, HRV and respiratory rate are selected and directly stored in the PostgreSQL database into their respective tables. Additionally, the RR-intervals themselves are also stored in a database table. When a mood is logged by the user in the mobile application, stored RR-intervals from the first three preceding minutes are retrieved. These are used for calculating the heart rate features used to train the machine learning model.

In Chapter 2, it was mentioned how an estimation of the breathing rate can be derived from RR-intervals. Because no breathing sensor could be used in this thesis, this method serves as a substitute. The method to estimate the respiratory rate is taken from the `heartpy` library. First, the method generates an array of x-coordinates (`x`) ranging from 0 to the length of the RR-interval list. The array itself is specified to be the same length as the RR-interval list. Next, another array with x-coordinates is generated (`x_new`). The coordinates also range from 0 to the length of the RR-interval list, but the length is specified to be equal to the sum of the RR-intervals, which creates a higher temporal resolution needed to analyse the breathing frequency within the time of the heart rate recording. A smoothing 2D-spline is fitted to the list of RR-intervals with array `x` as its coordinates, which will then be used to upsample the RR-intervals to 1000hz according to the coordinates of `x_new`. Next, the RR-intervals are passed through a bandwidth filter of 0.1hz to 0.4hz, corresponding to the high frequency HRV band. Finally, the Fast Fourier transformation is applied for spectral density analysis to get all the frequency powers residing withing the filtered signal. The highest power obtained is used as the estimated breathing frequency. This value multiplied by 60 is equal to the estimated breaths per minute.

```python
    #Resample to 1000hz
    x = np.linspace(0, len(rrlist), len(rrlist))
    x_new = np.linspace(0, len(rrlist), np.sum(rrlist,
        dtype=np.int32))
    interp = UnivariateSpline(x, rrlist, k=3)
    breathing = interp(x_new)

    #Filter signal
    breathing = hp.filtering.filter_signal(breathing,
        cutoff=bw_cutoff, sample_rate = 1000.0,
        filtertype='bandpass')

    #PSD analysis
    datalen = len(breathing)
    frq = np.fft.fftfreq(datalen, d=((1/1000.0)))
    frq = frq[range(int(datalen/2))]
    Y = np.fft.fft(breathing)/datalen
    Y = Y[range(int(datalen/2))]
    psd = np.power(np.abs(Y), 2)

    measures['breathingrate'] = frq[np.argmax(psd)]
```

Code Fragment 5.2: Heartpy python algorithm to estimate breathing rate

It is important to know that this estimation does reflect the real process of inspiration and expiration of the lungs, but rather the effect of an autonomic breathing process on the heart under normal circumstances. Inhaling and exhaling many times on purpose would therefore not increase the breathing rate. While this estimation may sound like a more accurate measurement of how a person really reacts to stimuli, the results aren't always accurate. For example, it seems to fail capturing high respiratory rates during physical exercise. Because exercise activates the sympathetic nervous system, this can be attributed to influences on the heart that are not detectable in the high frequency range, which reflects parasympathetic activity. Nevertheless, it is interesting to be able to measure the interplay between breathing rate and the heart.

5.3 Stress predictor

To predict stress based on heart rate features, the k-nearest neighbours (kNN) algorithm is applied for classification. It is a supervised learning technique, meaning that a labeled set of training data is used to predict outcomes and recognize patterns. During initialization, the kNN algorithm stores the training data in an F-dimensional space, where F is the number of features. The algorithm works in two stages: the determination of the k nearest neighbours of the sample by using a distance metric, and determining the class of the sample once they are obtained. To classify a new sample, the algorithm simply looks at the k nearest data points in the F-dimensional space in order to predict the class. When $k = 1$, the algorithm simply looks at the class of the nearest data point to the sample. With $k > 1$, the algorithm chooses through majority voting with uniform weights, or by a distance weighted voting mechanism. Figure 5.8 shows an example of kNN neighbour selection in a 2D space. The icon with the question mark represents the new sample to be classified, and the circles and triangles represent training samples of two classes. It becomes clear that in a majority voting approach, the new sample will be classified as one of the blue circles in the cases of $k = 3$ and $k = 7$ because they outnumber the red triangles.

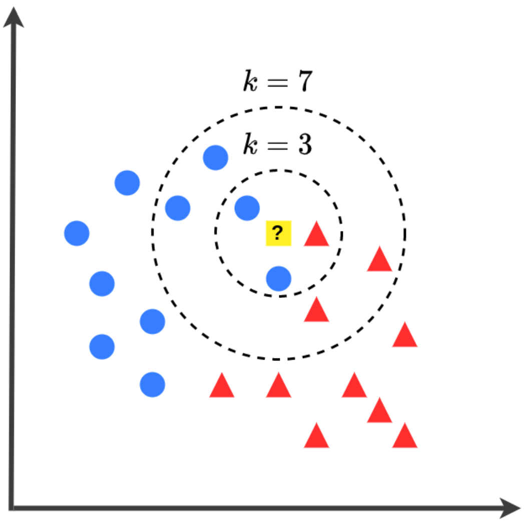


Figure 5.8: kNN with different values of k [34]

The equation below shows the distance d between a new sample q and a training sample $(x_i)_{i\in[1,n]}$ out of n total samples in the training set. $\delta(q_f, x_{i,f})$ represents the chosen distance or similarity metric to calculate the distance between sample q and x_i in the fth dimension, with w_f being the weight for the feature in this dimension. kNN bears similarities to its unsupervised learning counterpart called k-Means, which is used for clustering data points into groups based on feature similarity [11]

$$d(q, x_i) = \textstyle\sum_{f\in F} w_f \delta(q_f, x_{i,f})$$

Because of the spacial interplay between different heart rate features, the chosen distance metric of choice in this thesis is the Euclidean distance. This is the straight-line distance between two points in Euclidean space. The magnitude of each coordinate pair in dimension f is summed and squared, resulting in the Euclidean distance between points q and x_i. The formula becomes as following:

$$d(q, x_i) = \sqrt{\textstyle\sum_{f\in F} w_f (q_f - x_{i,f})^2}$$

Because training examples must be present in memory during run-time, this approach is often referred to as Memory-Based Classification. This technique is also categorized as Lazy Learning because the process of finding patterns in the data is deferred until run-time. In this thesis, this is considered an advantage. By logging the mood in the mobile application, the kNN model is 'trained' by simply adding another row of features to the training data. This is in contrast with other machine learning techniques such as a neural network, in which the model needs to undergo a training stage in order to be able to make predictions. The approach of continuously training the model as new data becomes available is also known as online learning.

Being able to detect stress from ECG data for the general population would be quite a task. It requires gathering a substantial amount of data from diverse population groups and considering their unique characteristics. Patterns of stress could either be detected for each specific group by applying stratified sampling, or a common pattern across all population group could be detected. Either way, it requires lots of data. Many of the datasets used in existing research are not disclosed for the public due to privacy reasons, and gathering the data ourselves is something which wouldn't be feasible for this thesis. The solution proposed in this thesis is to let the user log his mood to continuously train a model based on his individual physiology. The individual model approach, however, comes with a cold start. A common approach is to let the user log subjective feedback for a specific amount of time before letting the machine learning algorithm perform predictions. Devices such as the Pixel Watch are already trained on some data from studies to test their machine learning algorithms, and this data is complemented by additional user data. In this thesis, a similar approach is taken.

Training data is taken from the ECG subset of the WESAD dataset. Since this dataset is comprised of data collected from young and healthy students, just like the author of this thesis, it is a suitable dataset to train and test the performance of the kNN model with. Furthermore, the notion of stress defined by the WESAD study by using the Trier Social Stress Test (TSST) corresponds to the idea of stress that was in mind for this thesis: not mild annoyances as defined by other public datasets, but real stress. As defined by the problem statement of this thesis, it is also essential to detect signs of physical stress. Because data labeled with physical stress is not provided by the WESAD dataset, RR-interval data samples from intense exercise as visualized in Chapter 4 are added to the training data and labeled with physical stress, amounting to four hours worth of data by using the sliding window approach. The amusement label is replaced with baseline to simplify the classification process. The Python library used to create the model is `scikit-learn`. Parameter k is chosen at the recommended default setting of 3. Majority voting is chosen by default and no weights are assigned to the data points. The WESAD dataset contains 62 heart rate features in total, but most of them are simply the normalized versions or logarithms taken of other features. Hence, most of the features were to prevent redundancy and increased complexity. The following 18 features are retained for training:

Table 5.2: Description of HRV Variables

Feature	Description
MEAN_RR	The mean RR-interval value in milliseconds.
MEDIAN_RR	The median RR-interval value in milliseconds.
SDRR	The standard deviation of RR-intervals.
SDSD	The standard deviation of differences between successive RR-intervals.
RMSSD	The root mean square of successive differences in RR-intervals.
HR	The heart rate in beats per minute.
pNN25	The percentage of adjacent RR-intervals whose difference is higher than 25 ms.
pNN50	The percentage of adjacent RR-intervals whose difference is higher than 50 ms.
SD1	The standard deviation of points perpendicular to the line of identity on a Poincaré plot.
SD2	The standard deviation of points along the line of identity on a Poincaré plot.
VLF	The spectral density power of RR-intervals in the very low frequency band (0.0033 Hz - 0.04 Hz).
LF	The spectral density power of RR-intervals in the low frequency band (0.04 Hz - 0.15 Hz).
HF	The spectral density power of RR-intervals in the high frequency band (0.15 Hz - 0.4 Hz).
LF_NU	The low frequency power divided by the total power.
HF_NU	The high frequency power divided by the total power.
TP	The total power (VLF + LF + HF).
LF_HF	The ratio of low frequency power to high frequency power.

After filtering out the most important features and adding the physical stress data, the performance of the model was tested by splitting the data into 80% training and 20% testing subsets. Figure 5.9 shows only a minor 0.01 percent decrease in accuracy. Judging from the very high accuracy, it becomes clear that the kNN model performs well on the test data. This is due to the fact that the training data mainly consists of very similar data points for each class. The question is: will the model also perform well on unseen data? The answer is sadly no. During many instances of wearing the chest band and using the mobile application, there was an overflow of false positives of stress being detected by the predictor. Mental stress data from the perceived stress logs was also added to the training data in order to attempt improving the accuracy of the model, but that didn't make much of a difference. This issue can be attributed to the lack of variation in the data, which makes the model not able to generalize well. It can also be partly due to the poor performance of the kNN model in comparison with other machine learning techniques, as mentioned in Chapter 3. But even though the model does not perform well, the system of continuously being able to make predictions about stress and train the model on new mood data works according to expectations.

Classification Report for Test Set:

	precision	recall	f1-score	support
baseline	0.99	0.99	0.99	18878
physicalstress	1.00	1.00	1.00	2913
stress	0.98	0.98	0.98	8240
accuracy			0.99	30031
macro avg	0.99	0.99	0.99	30031
weighted avg	0.99	0.99	0.99	30031

Figure 5.9: kNN performance tested on the test set [34]

The predictor itself also runs on a FastAPI application in order to communicate with the data processor. It contains two endpoints: one for training the model and one for making predictions on a new sample. Starting with initializing the API program, the kNN model will be trained on the data in the .csv file provided in the directory of the program. Afterwards, the model can simply make predictions when the data processor calls the prediction endpoint. When calling the training endpoint, a new row of features will be added to the .csv file, and the model will load the updated data into memory. To prevent potential timeouts on the endpoints, the training of the model is executed on a separate thread, making the endpoints open to new requests while the training process runs in the background.

The WESAD dataset also contains a subset of EDA features. Even though stress predictions could be made using these features, the Google pixel watch only allows exports of data in one minute intervals while the WESAD dataset contains data from each second of recording, presumably obtained through a sliding window approach. Due to the incompatibility in these formats, EDA-levels are not included in predicting stress. Even though a multi-modal approach is likely to be more accurate in detecting stress as it does not solely rely on the fluctuations in heart rate patterns, Giannakakis, Marias, & Tsiknakis (2019) [14] have shown to obtain promising results in detecting stress based on heart rate features alone.

5.4 Web application

To make users more aware of their physical and mental health, a web application was developed in order to visualize the data aqcuired by the user. The application is written in Javascript using the ReactJS library developed by Facebook. The focus of ReactJS lies in building reusable, component-based UIs where each component manages its own state, making it easier to develop and maintain complex user interfaces. React uses a virtual DOM (Document Object Model), updating only the necessary components when data changes. This is also called hot reloading. The first and main part of the web application is the interactive timeline. It is developed using Vis.js, a JavaScript library for visualizing data in the form of interactive timelines, networks and graphs. While a React version exists for this library, it is not used because it does not really provide any advantages. Using the standard Javascript library, there is better control over the initialized object because many layout aspects of the timeline needed to be tweaked with code. Two components of the library were used: Vis.Timeline and Vis.Graph2D. The Timeline component is used to create timelines, and visualize event data and their lengths with customizable rectangles. Users can zoom and drag within the timeline interface to select specific time intervals and to obtain more granularity. The Graph2D component is built on the Timeline, and also allows dragging and zooming, but is used for visualizing time series data in line charts. Figure 6.3.1 shows the end result of combining these two components. By stacking multiple Timeline and Graph2D components on top of each other and delegating zoom and drag events to other components when interacting with one component, it acts like a single unit.

When using the timeline, the user needs to select a specific day from which he wants to retrieve data. Having requested this data from the API, the script will then convert the json data into a Vis.Dataset object to feed the visualizations. During development, the date picker was developed to work for date ranges at first. However, the amount of data from multiple multiple days proved too much for the timeline too handle. Too many elements would be rendered into the timeline, causing the performance to decrease tremendously. There was an attempt at loading in the data each time the date range changed by zooming in or dragging. Each seperate row in the timeline would receive a trigger to get data and update itself, but this caused a lot of synchronization bugs. Zooming is performed by scrolling the mouse wheel, and dragging by holding in the left mouse button and moving the mouse left and right. The timeline consists of three main parts: the data logged by the user, the data calculated from the sensor measurements, and the stress predictions made by the machine learning model. Together, they form a whole to provide contextual information about the overall well-being of an individual. The line charts visualizing sensor data enable domain experts to pinpoint specific moments of changes in physiological signals, facilitating a more granular analysis. The heart rate features in particular are also meant to serve as an explanation as to why the machine learning model has predicted stress. What makes this timeline different from other visualizations is the direct comparison of physiological signals with other information over time.

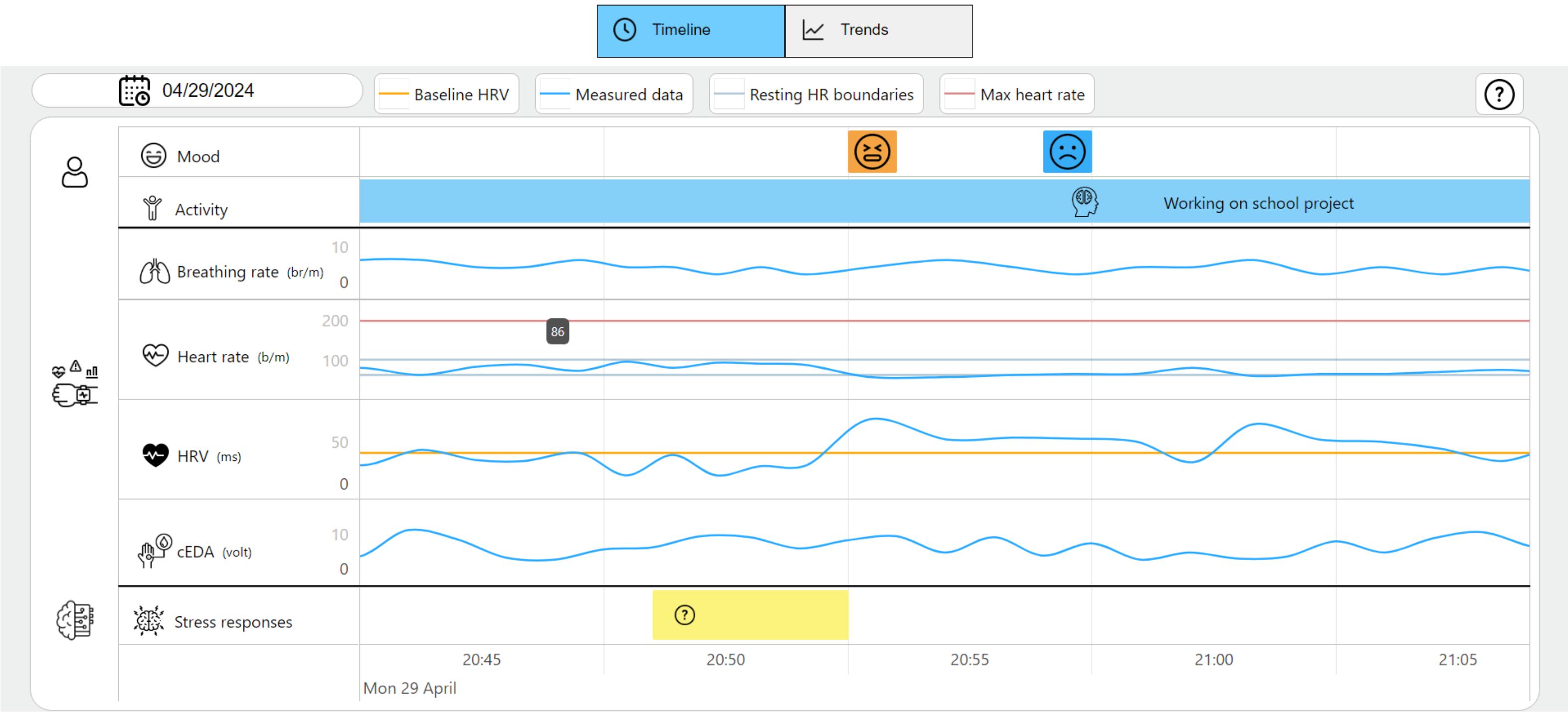


Figure 5.10: Web app timeline

Normally all lines would be connected with each other in the graph, unless they belong to different Vis groups. And by default, these different groups would be assigned a different color. Because no line needs to appear when no data is recorded, separate data groups are needed while maintaining the same style and name. As a solution, if data points are too far away from each other in time, a new group is created. This is achieved by maintaining a "group counter", which increments to create a new group for the next series of data, resulting in the separation of lines in Figure 5.11. The presence/lack of data is further emphasized through letting tooltips appear on hovering. The unique group names that normally appear in the legends of their respective graphs are added to the top of the timeline instead.

Figure 5.11: Data seperation and tooltip

Each logged mood is visualized with its corresponding emoticon and color. Clicking on one of the logged moods, for example the stress mood, will open a popup showing the mood type and the description the user has written in the mobile application, as shown in Figure 5.12:

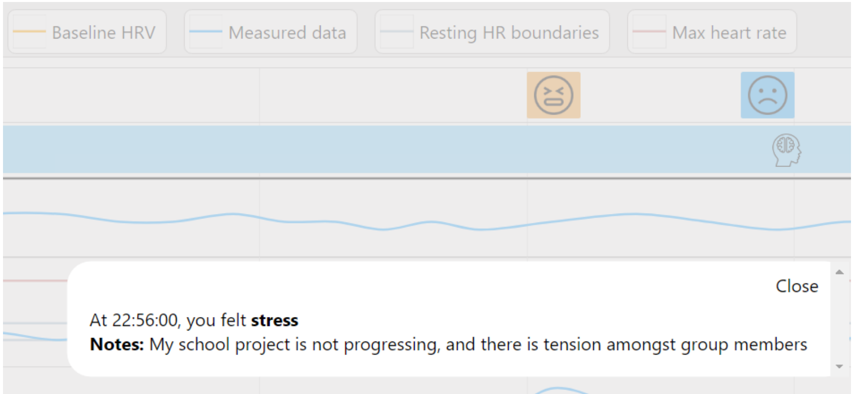


Figure 5.12: Mood information popup

When clicking on an activity, a popup showing its type and duration appears on top. Below, the physiological measurements during the activity are compared with the averages of those of activities of the same type. Take studying for example, which is a mental activity. A higher heart rate and lower HRV might indicate a more difficult time than other sessions, such as studying the day before a hard exam.

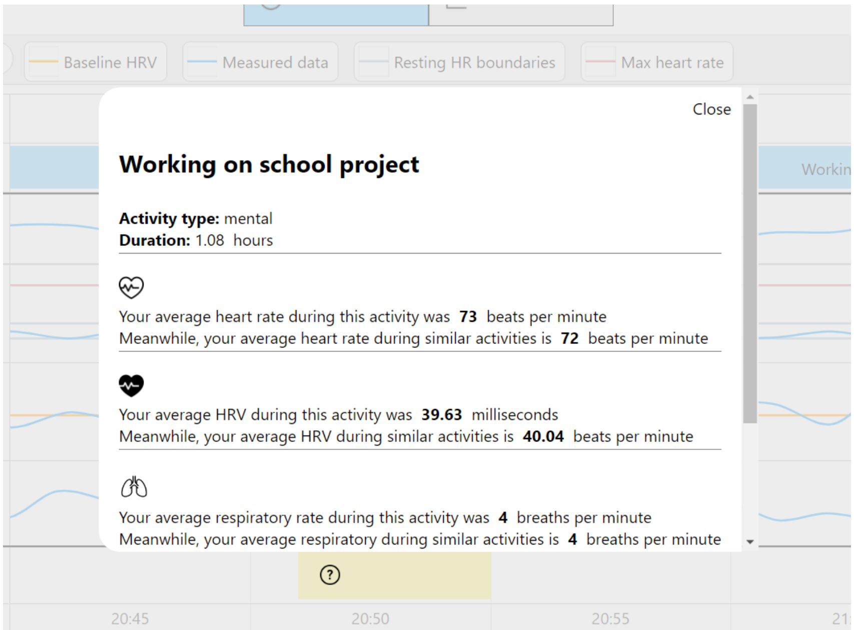


Figure 5.13: Activity information popup

On the top right of the timeline, there is a button with a question mark located. Clicking it will open the popup shown in Figure 5.14. This popup serves to inform the user about some of the medical aspects behind the rows in the timeline. Each medical claim is backed up with a source, which can be found by scrolling down to the bottom of the window.

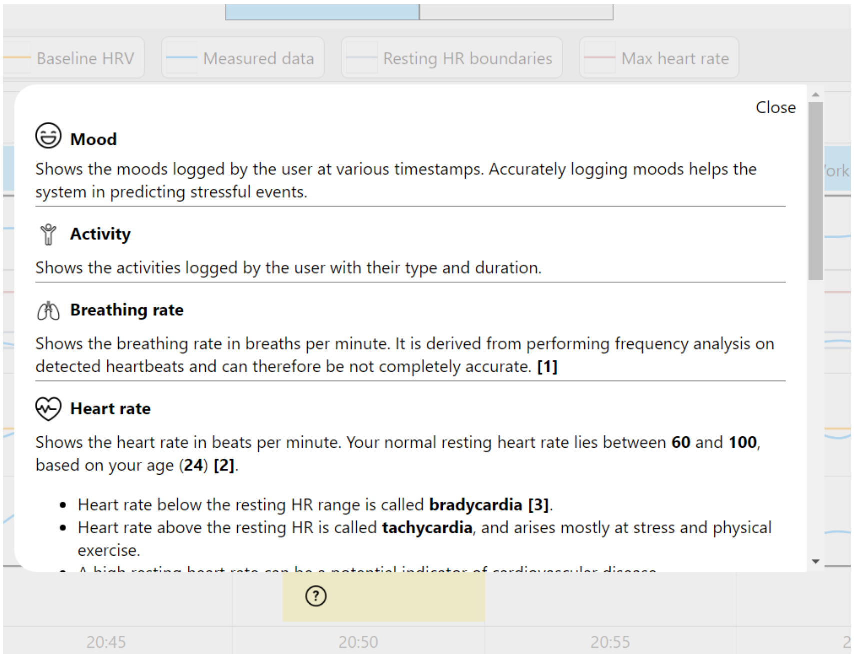


Figure 5.14: Background information screen popup

Clicking on an event in which the machine learning model has detected stress, the popup in Figure 5.15 will appear. This window will give additional information about the heart rate and HRV levels during the time of prediction along with advice to cope with the detected type of stress. Usually a stressful event is accompanied by a higher heart rate and lower HRV level, making it two of the most decisive features for predicting stress. Showing when the HRV drops below the baseline and how much the average and maximum heart rate compare to normal levels is meant to provide more transparency as to why the machine learning model detected the things it did. This approach is similar to feature-based explanations in Explainable AI.

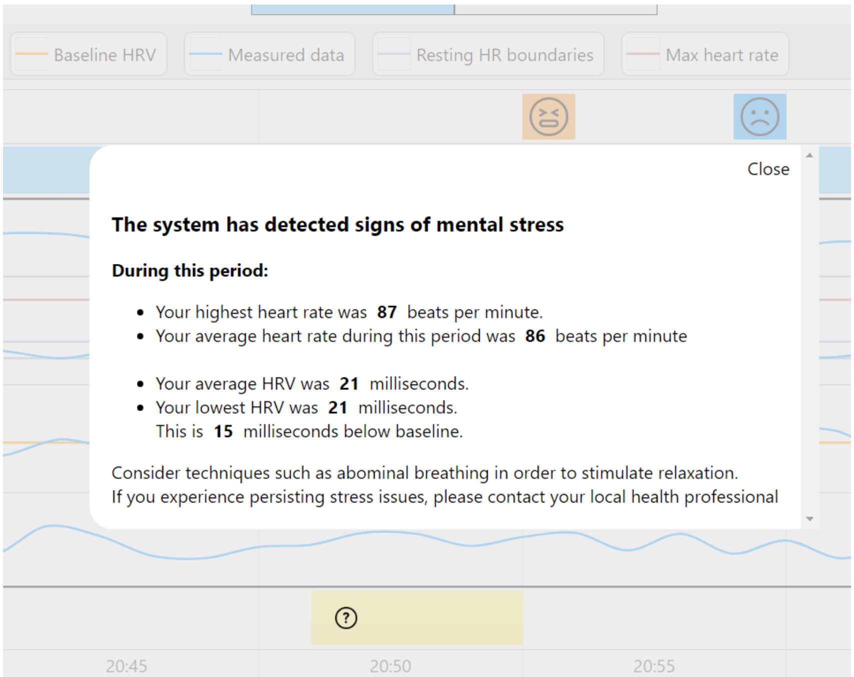


Figure 5.15: Background information screen popup

The second part of the application is the trends page. This page is meant to show the evolution of metrics related to overall health and stress management from the last x days, weeks or months in a grid-like layout via multiple plots. A change in the number selector or dropdown will automatically trigger a re-render, updating the graphs immediately. The first plot shown in figure 5.16 shows the amount of logged moods along with their types from six weeks ago until the current date. The colors in the bar chart are the same as the ones in the mood timeline for consistency. The point of this bar chart is to make the user aware of the 'progress' he has made in bettering his mental health, but also partly his physical health by showing the amount of moments of exhaustion logged by the user. All the charts are created using the ReactECharts library, which is a React version of the standard Apache Echarts library.

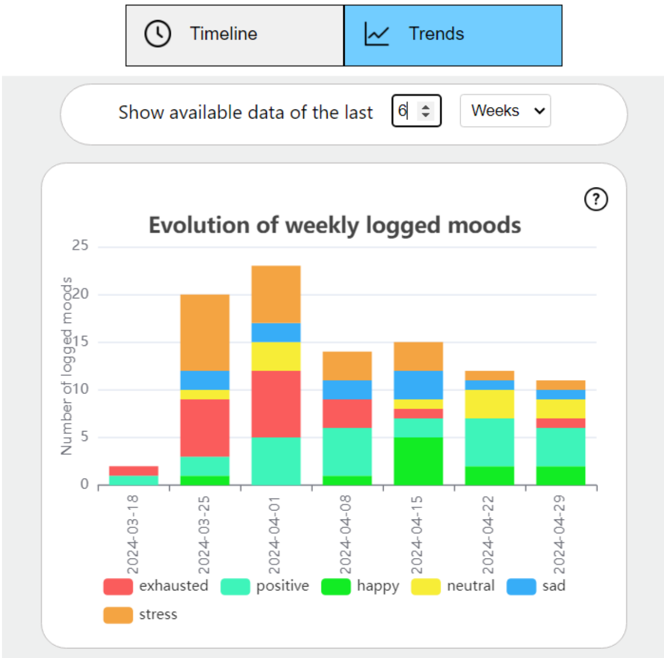


Figure 5.16: Evolution of logged moods

The next is row shown in Figure 5.17, in which the evolution of physiological metrics are shown in the chosen time range. The first plot shows the evolution of average heart rate. A higher average heart rate at one day might indicate a more stressful day, but this ultimately depends on when the sensor was worn and for how long. If worn consistently throughout the entire day, a lowering trend in average heart rate might indicate a lower resting heart rate, showing an improved cardiovascular system. The same thing applies for the HRV plot. An increase in HRV might indicate improved resilience of the parasympathetic nervous system for handling stress, but it ultimately depends on when the user wore the heart rate sensor.

Figure 5.17: Evolution of hr and HRV

Additional evolution is shown in Figure 5.18. The first plot shows the amount of stress responses detected by the system. The more downwards the line in the chart goes, the less signs of stress the system has predicted. A lowering trend would indicate a better adaption to stress, or just less stressful moments experienced throughout daily life. The second plot shows an evolution of the amount of physical activity spent in hours. Of course, a rising trend indicates an increase in physical activity, meaning that the individual put in more effort in staying physically active.

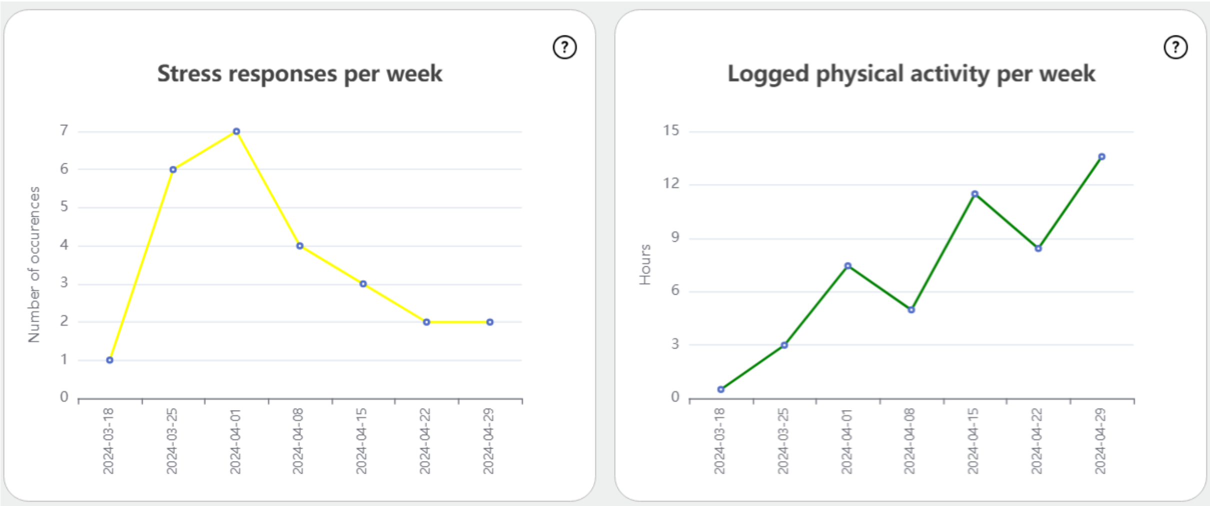


Figure 5.18: Evolution of stress responses and physical activity

With these charts put together instead of viewed separately, people are able to notice the correlations between them. A decrease in stress responses can for example be attributed to the increased amount of physical activity spent by the user. Going for walks or training in the gym, for example, is generally known to be way to cope with stress. Furthermore, an increased amount of physical activity can also be reflected in the logged moods. In the case of students for example, studies have shown more physical activity grants strengthened memory, better concentration and increased energy levels [1]. The downside of this visualization is that when the individual only wears the sensor during physical activities, heart rate will be shown to be high on the graphs, and HRV will be shown be on the lower side. Hence, because of this uncertainty, the word *may* is carefully used in the help screen when clicking on the question mark button above the graph. Figure 5.19 shows an example of clicking on the button above the heart rate graph.

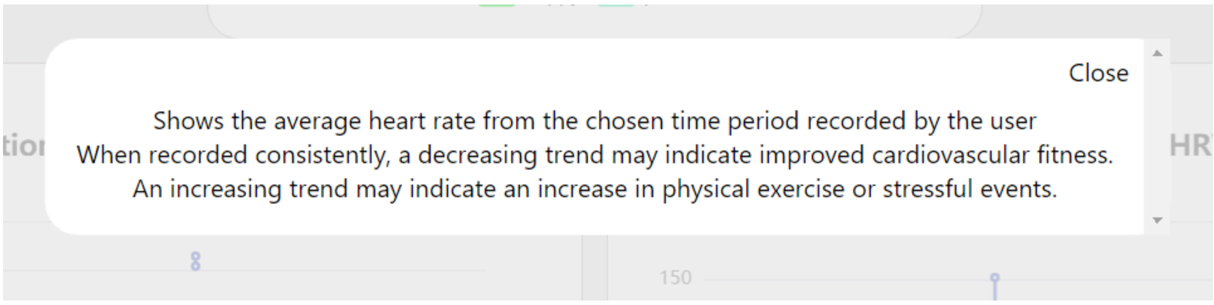


Figure 5.19: Graph help window

In all of the line charts above, the points of recorded data are shown. By consulting the timeline, the user has the opportunity to determine the exact causes of why the trends are shown as they are, for example where data was missing and at what timestamps the sensor was worn.

Chapter 6

Usability tests

Having developed a prototype system for stress and health management, it is essential to measure how effective it is when used by others. In this thesis, the focus is solely on studying the usage of the web application. Evaluating the effectiveness of the stress detection algorithm while wearing the Polar heart rate sensor and using the mobile application is hindered by the large amount of false positives in detected stress responses and other resource constraints. Therefore, the objective of this study is to assess the usability aspects of the web application and evaluate the comprehensibility of the data presented in the visualizations, quantifying satisfaction through both scores and additional textual feedback. After analysing the results of the study, it can be determined how useful the timeline approach was in contextualizing physiological signals with subjective feedback.

6.1 Method

6.1.1 Recruitment

Participants must be at least 17 years old to comply with legal regulations concerning data privacy and consent [9]. No other specific criteria are required to participate, as monitoring stress and health is an activity open to the general population. A total of 19 participants, who are personally acquainted, are enlisted through direct messaging via platforms like Facebook and Discord, as well as through face-to-face interaction.

6.1.2 Setup and procedure

During the tests, the actions of the user are recorded. This is done in order to time the usage, and to assess any mistakes made by the participant to prevent them being overlooked by the observer. A think-aloud principle is used, which means that the user has to continuously talk about the actions they perform. With this principle, their vocal reasoning is recorded during the process of using the web application. This can help in assessing how straightforward the user interface is and in what aspects it is lacking.

The study itself takes place both in the HCI demo room of the EDM and at the participant's homes in a quiet room. To prevent any disturbances, a sign is hung on the door outside. The participant is welcomed by the supervisor and is asked to take a seat in front of a prepared laptop at the table. Initially, the participant is asked to read a sheet containing the purpose of the study and how it will be conducted, and then read and sign a consent form. If the participant declines, they are thanked and guided outside. If the participant agrees to sign, they are given a sheet with information that is required to evaluate the web application. This sheet contains the description of a person that has hypothetically made use of the mobile application before in order to record physiological data and submit feedback through logged moods and activities. It describes two types of instances in which this person wants to recall certain stressful events by using the web application to determine the causes of and correlations to the experienced stress. Furthermore, it also describes one instance in which this person wants to determine how his overall health has progressed. The contents of the sheet are shown in English in section 6.1.4. After reading the information sheet, the participant is once again reminded of the think-aloud principle, and the recording is started. During the usage of the web application, the user is allowed to consult the information sheet. If the user is stuck for a long time or has complaints, the observer will interfere. Otherwise, the observer will keep a distance to not make the participant nervous. After completing all the instances, the recording is stopped and the user is asked to fill in an online questionnaire.

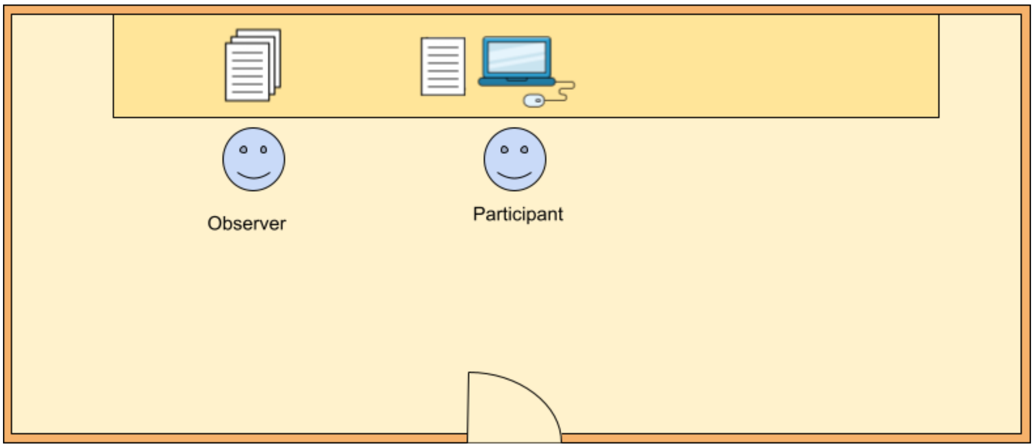


Figure 6.1: Sketch of the testing environment

In preparation of the real study, a pilot study with Professor dr. Ruiz was conducted first. The feedback from the professor was used to update the web application and the documents provided in the study as necessary, as well as how the participant is guided throughout the study.

6.1.3 Materials and artefacts

The usability study is conducted with a personal Acer laptop and a mouse. The screen and voice recordings during the user study are performed using the OBS screen capture software. The questionnaires are made using Google forms. This way, data can be collected automatically in a spreadsheet, which is linked with these forms. The google chrome web browser is used to run the web application. To use the application, it is mandatory that the user uses the provided mouse, as zooming in the timeline can cause zooming into the canvas.

6.1.4 Participant objective

The contents on the information sheet which the participant uses to navigate through the web application are as follows:

Person Description

- Man
- 24 years old
- Student

Previously, this fictional person collected data using wearable sensors and a mobile application. The following scenarios describe the information this person wishes to request using the web application.

Instance 1

“Last week on Monday, I felt mentally unwell in the evening. At the moment, I cannot remember any specific event or trigger that could have influenced my mood. Therefore, I would like to gain more insight into why I felt this way and what the possible causes could have been by querying my logged moods and activities."

Instance 2

“Last week on Wednesday, I went squatting, but it was very hard for me. Even an hour after the session, I still felt exhausted. Normally, I feel refreshed after squatting, but this time I still felt very fatigued after my session. It felt like my body had more difficulty than usual, and I was concerned about my performance and well-being. I would like to check my heart rate during this period to see if there is possibly a connection between my fatigue and other factors during my gym session."

Instance 3

"In recent months, I often found myself in a state of unhappiness and regularly felt tired. For this reason, I decided seven weeks ago to improve myself and actively work on my mental and physical well-being. At this moment, I am curious about the progress I have made. I would like to check how successful I have been in improving my overall well-being."

In order to be able to measure the amount of errors the participants make in the navigation and thinking process, it is essential to pre-establish a 'correct' flow when trying to retrieve information for each of the three instances. That way, we can differentiate the expected usage from the actual usage of the application. As for the domain expert or other participants which happen to have medical knowledge, additional meaning might be derived from the measurements that were overlooked during the previous heart rate analysis. For each instance, the ideal flow is as follows:

Flow instance 1

1. Upon opening the web application, the user finds the correct date of recording using the date picker. In this instance, it is last week's Monday.
2. Having chosen the correct date, the timeline will give an overview of the entire day of last Monday. Because the stressful event took place in the evening, the user uses the dragging and zooming with the mouse to gain better resolution of the data.
3. The user explores each row of the timeline and its features. These correspond to images 6.3.1 - 5.15 in the previous section. Exploring the rows on the timeline does not have to be in order.
    - From the mood row, the user notices two logs: a stressed one that indicates trouble with a school project, and a sad one that indicates unfulfillment in life. Based on these logs, the user should understand why the person has experienced stress.
    - The user clicks on the element in the activity row, and notices that there are not really any anomalies in the comparison sections.
    - From the metrics, the user notices that the HRV falls below baseline during some instances and that there are some spikes in heart rate. For the rest of the metrics, the user sees no particular anomalies. Because there was much ambiguity in the heart rate features during previous analysis of mental stress, it is expected that the logged moods are the most decisive in finding out why the person felt the way he did.
    - The user clicks on the stress response and tries finding out why the system detected stress.
    - If the user does not understand particular metrics, he/she will consult the help screen.

Flow instance 2

1. After finishing instance 1, the user navigates to Wednesday in the same week. This can again be done with the date picker.
2. Having chosen the correct date, the timeline will give an overview of the entire day of last Wednesday. Two sequences of recorded data will be shown. Because the stressful event took place during the second sequence in the afternoon, the user uses the dragging and zooming with the mouse to gain better resolution of that data. This is shown in Figure 6.2.
3. The user explores each row of the timeline and its features. Again, this does not have to be in order.
    - From the mood row, the user notices two logs: a neutral one indicating a feeling of lightness in the head, and an exhausted one which says that the person felt weak due to digestion while going heavy with lifting weights.
    - When clicking on the activity, the user sees no particular difference in averages compared to similar activities.
    - From the metrics however, it becomes clear that the user underwent training as the heart rate is elevated for a long time whereas HRV is below baseline for a long time, as well as higher levels

electrodermal activity. This is also shown in the stress response window, which the user checks. The user understands these phenomena are inherent to physical exercise.

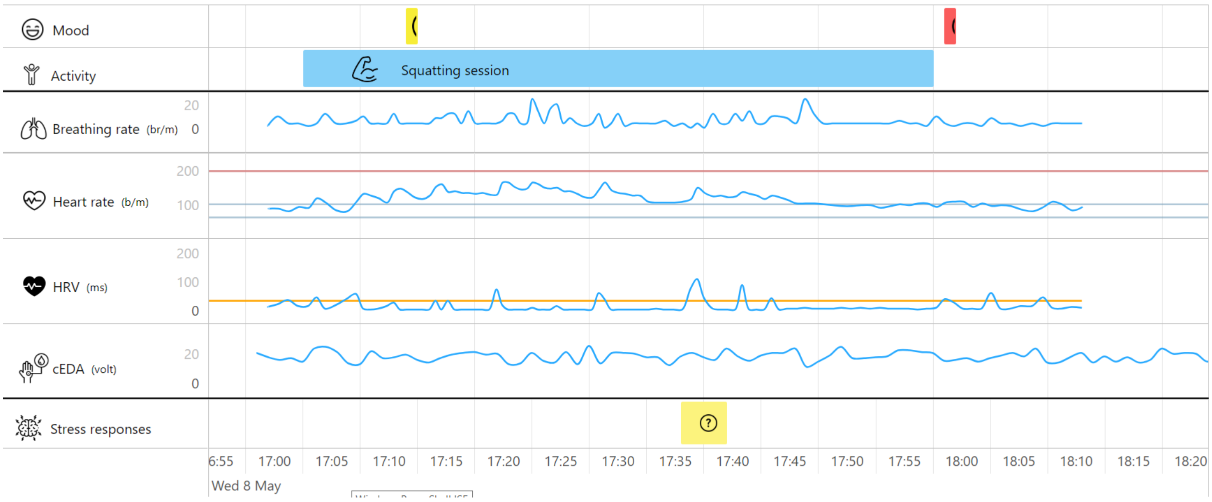


Figure 6.2: Stress log of squatting on wednesday

Flow instance 3

1. After finishing instance 2, the user navigates to the trends tab in order to see the person's evolution in mental and physical health. The features in this tab correspond to Figure 5.16 - 5.19.
2. With the time unit and number picker, the user preferably chooses to retrieve data from the last 7 weeks as this gives the best resolution into this evolution.
   - From the bar chart showing the different types of logged moods and their amount, the user concludes that there has been a significant reduction in stress and exhaustion logs and a slight increase in positive logs, and thus the person has succeeded in improving his mental health.
   - From the heart rate graph, the user notices the rising trend. When the heart rate is always monitored, this may indicate a problem. However, the person has been performing more physical activity. The user realises that this increase is due to the recording of the heart rate prominently during the events of physical activity.
   - The user notices a dip in the HRV graph, which can also be attributed to the increase in physical activity. Before the setup of the user studies, no conclusions could be taken as to why it rises again afterwards.
   - From the stress responses graph, the user clearly sees a consistent downwards trend. From the physical activity graph, the user sees an upwards trend. The user concludes that the person has been doing well in improving his physical health. As sports can be seen as an outlet to deal with stress and negative emotions, whether this may or may not have helped him doing better mentally is up to the user's interpretation.

6.2 Results

6.2.1 User performance & errors analysis

For each participant, interesting observations/remarks made by the participant are noted in the Appendix, as well as the amount of flow errors made by the participant and software bugs discovered by the participant. Table 6.1 below summarizes the performance all participants.

Table 6.1: Summary of participant performance

Participant	Number of Errors	Number of Bugs Found	Duration
Participant 1	1	0	10:22

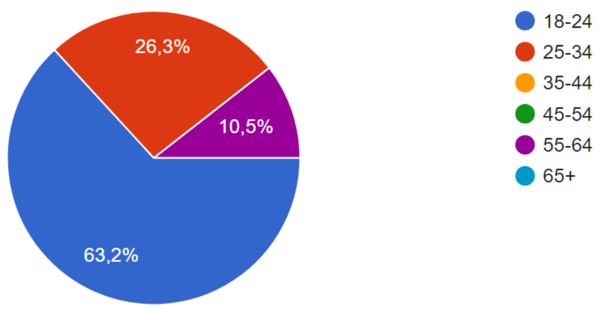
Table 6.1 Summary of participant performance – continued

Participant	Number of Errors	Number of Bugs Found	Duration
Participant 2	3	0	06:18
Participant 3	2	2	09:39
Participant 4	4	0	04:08
Participant 5	3	0	10:37
Participant 6	3	0	5:32
Participant 7	1	0	08:11
Participant 8	2	0	06:20
Participant 9	3	0	18:50
Participant 10	1	0	10:46
Participant 11	1	1	15:57
Participant 12	2	1	18:04
Participant 13	4	0	10:57
Participant 14	3	0	12:18
Participant 15	3	1	16:34
Participant 16	2	3	12:53
Participant 17	5	0	11:43
Participant 18	1	0	5:43
Participant 19	3	1	12:42

6.2.2 Questionnaire feedback

In this subsection, the form submitted by the users after testing the web application will be discussed and the text answers will be summarized in bullet points. The form has a total of 21 questions. It comprises of yes and no questions with optional follow-up questions to allow clarification, open-ended questions for textual feedback and Likert scale questions for quantifiable feedback on opinions and user experience.

Question 1: what is your age category?



For transparency, it is important to mention the distribution of age groups between the 19 participants. The vast majority of people were individuals around student age due to the accessibility of fellow students and acquaintances around those ages. While age can be a determining factor in assessing the usage of a web application, it is not very relevant in this study. The focus of the application lies in interpreting and understanding displayed information, which is not significantly impacted by age. Secondly, doing a clear analysis between age groups would require far more participants.

Question 2: what are your initial thoughts about the concept and the purpose of the application?

Positive feedback

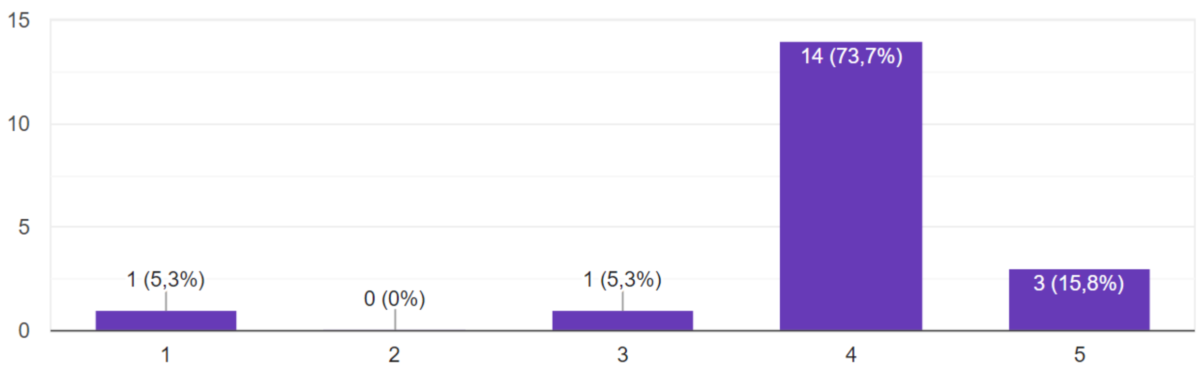
- The application has a general contribution to daily life. It has the possibility to track and evaluate physical and mental well-being. It is useful in emergencies such as heart problems for caregivers to get more context. Informative info bars help in understanding the data.

- Different metrics make it useful to 'browse' through the data. This also gives a better insight in the long term. Useful to see all the information inside one single application. It is overall a cool concept.

Critiques

- It is sometimes difficult to navigate, but eventually it's possible after finding out how it works. Some elements could be improved for a better user experience. It was not always clear whether some elements were clickable or not.
- It is difficult to directly link mental well-being with physical well-being. For people without medical expertise, interpretation of the displayed information can be a bit difficult.
- Measurements need to be continuous to gain good insights, which can sometimes be difficult for people. Detecting trends on their own could be difficult for some users. Data needs to be acquired over a long period of time to make the visualizations really useful.

Question 3: how useful/interesting did you find the features of the application?



In this range, 1 equals to very useless whereas 5 equals to very useful. 89.5% of the participants found the features of the application useful for stress management.

Question 4 (optional): which features did you find the most or/and least useful/interesting?

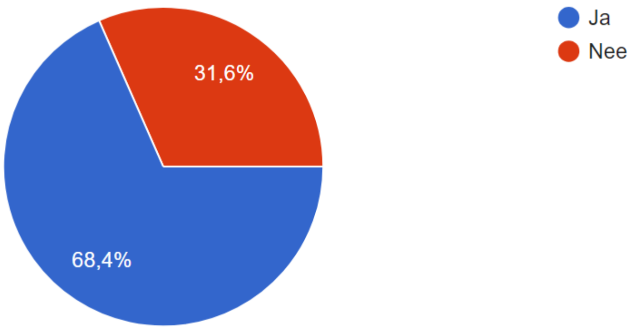
Positive feedback

- The timeline provides an overview of mental and physical well-being over a long period of time. It gives a quick and user friendly overview of different metrics with the zoom and drag function.
- The display of cohesion between the activities and the metrics belonging to them is interesting.
- The trends shows if taken actions to improve health had effect on the long term.
- An overview of detected stress responses was seen as useful.

Critiques

- Participants did not directly understand all the displayed terms.
- The app is dependent on the intensity of usage of the user.
- The breathing rate and cEDA were not really useful in context of the instances in the conducted study.
- The graphs don't tell much without medical knowledge.

Question 5: Were there any features that were unnecessary or could be improved?

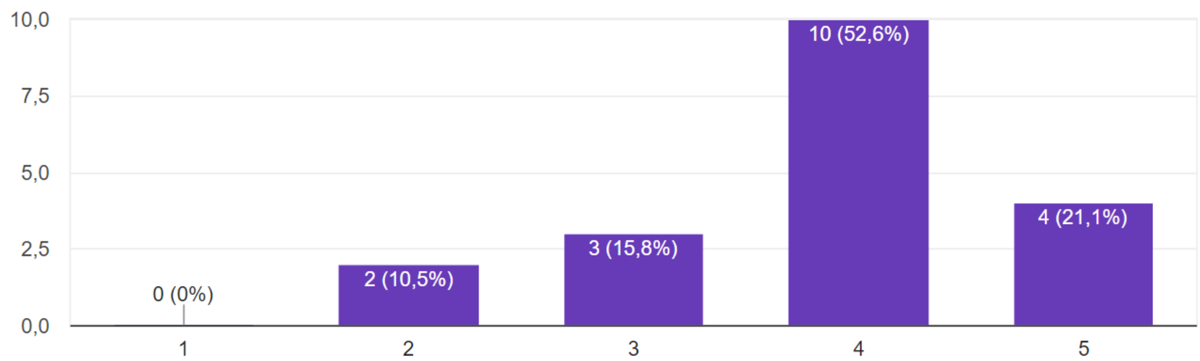


More than 30% of the participants think the features in the application could use improvement. This is quite a substantial amount, so it is necessary to identify these shortcomings by letting the participant specify these features in the follow-up question.

Question 6 (optional): In case there were any, which ones did you have in mind?

- The application needs more tooltips for the timeline and other functions to provide for more clarification. Make the help button more visible, and add more visual indicators and colors. Allow for clicking on the timeline metric labels for more information. Maybe add a wizard which briefly explains where the help button is located.
- Add more consistency in colors throughout the visualizations.
- Increase the size of the timeline tab and the graphs.
- Add a more simple text explanation with the medical terms.
- Less abstraction in the heart rate values.
- Add population-normal or expected values for personal comparison. Show trends relatively alongside absolute values to display percentage changes.
- Fix UI/UX flaws (American calendar/date standards are shown instead of European ones, long load time, no floating close button..)

Question 7: how easy did you find it to navigate through the timeline?

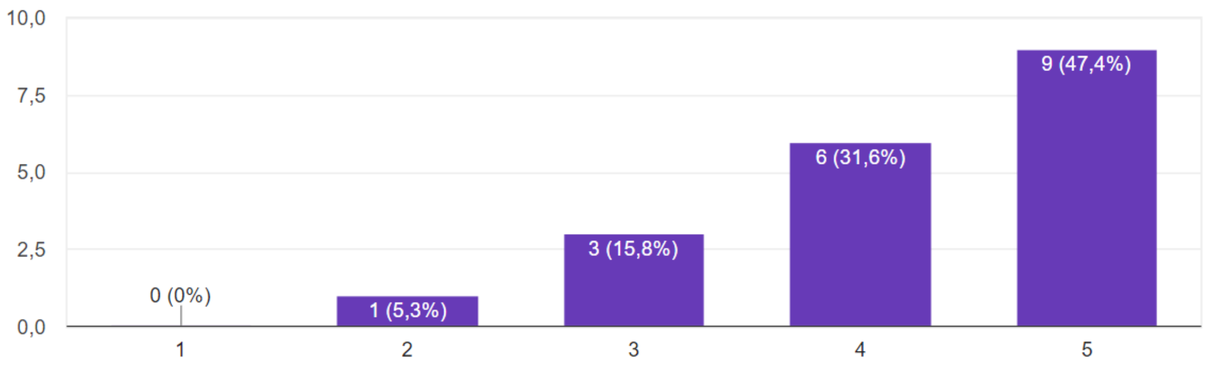


In this range, 1 equals to very difficult whereas 5 equals to very easy. Despite the timeline being a bit laggy, 73.7% of the participants found it easy to navigate. The cause of this lagginess is explained in the next section.

Question 8 (optional): in case navigating through the timeline wasn't easy, what were the difficulties you experienced?

- The timeline was a bit clunky, in particular the zooming and dragging. When zooming in, items suddenly seemed to disappear. Therefore, it was avoided sometimes later on.
- 'Scrolling' was not intuitive. Participants tried to go left and right with the mouse button, which was meant for zooming. A scroll bar is missing below.
- The calendar started on Sunday while Monday is preferred.

Question 9: how clear does the application present the data (logged moods, activities, metrics..)?

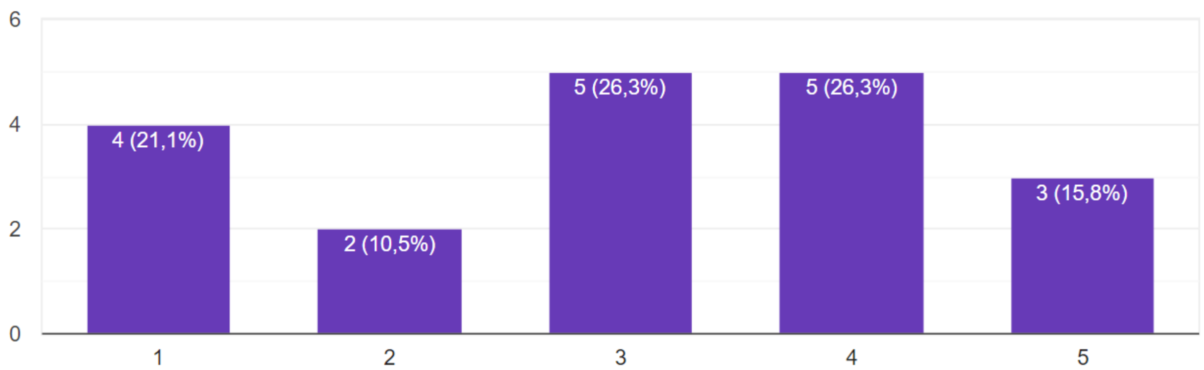


In this range, 1 equals to very unclear whereas 5 equals to very clear. 76% of the participants found that the application displays the data well.

Question 10 (optional): if there were any uncertainties, which were these?

- The button for the help screen wasn't easy to find. Rather, tooltips were expected when hovering over the labels in the timeline rows. When the participants didn't find the meaning of abbreviations like HRV, it caused confusion.
- In the trends page, the causes of the moods are missing and could be shown in a tag.
- The mood color for stress was mistaken for the line color for stress responses by the system.
- It was unclear why the moods had an arbitrary interval.

Question 11: in the application, some medical terms were mentioned to clarify the meanings behind the data in the timeline. For this information, you had the option to consult a help screen. How helpful did you find this screen?

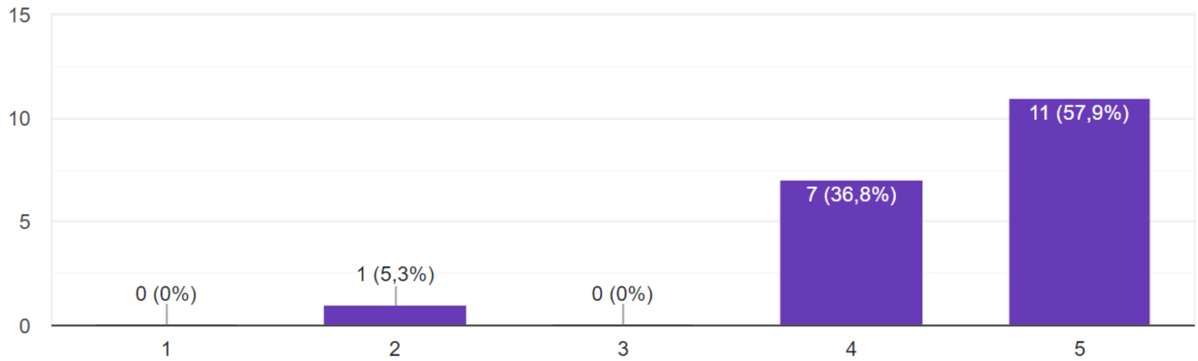


In this range, 1 equals to very unhelpful whereas 5 equals to very helpful. As expected from the inability to notice the button and the remarks of finding the medical explanations confusing during the recordings, the score on this question is divided. Less than half the participants found the help screen helpful, so the way of explaining the metrics needs revision.

Question 12 (optional): in case there were any uncertainties: which were these?

- The button to open the screen wasn't noticed by many. For some, it happened to be useful after it was pointing out the screen exists. Others considered it too difficult to interpret correctly.
- Some participants opted for a start with a simple explanation (e.g via tooltip), accompanied with an option to retrieve more in-depth info.

Question 13: in addition to logging moods and activities, to what extent do you find it valuable that the system can automatically detect signs of stress?

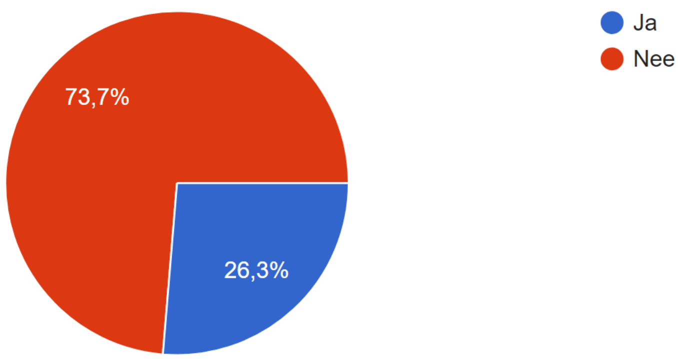


In this range, 1 equals to no added value whereas 5 equals to a lot of added value. Nearly every participant found the integration of automatic stress detection a positive addition to the system.

Question 14 (optional): why do you think it provides added value or why do you think it doesn't?

- Automatic stress detection can highlight underlying stress when an individual is not feeling well, something which can't always be noticed directly. People do not always realize how stressed they actually are; automatic stress detection helps with this. This also allows people to take timely action if necessary.
- Automatising saves time and effort because users don't need to log patterns themselves. It also prevents extra stress from manually logging stressful moments.
- In combination with logged activities, it helps users find out what exactly caused a stress response. How metrics differ from normal values during stress responses is also an advantage, because the user doesn't need to manually compare them.

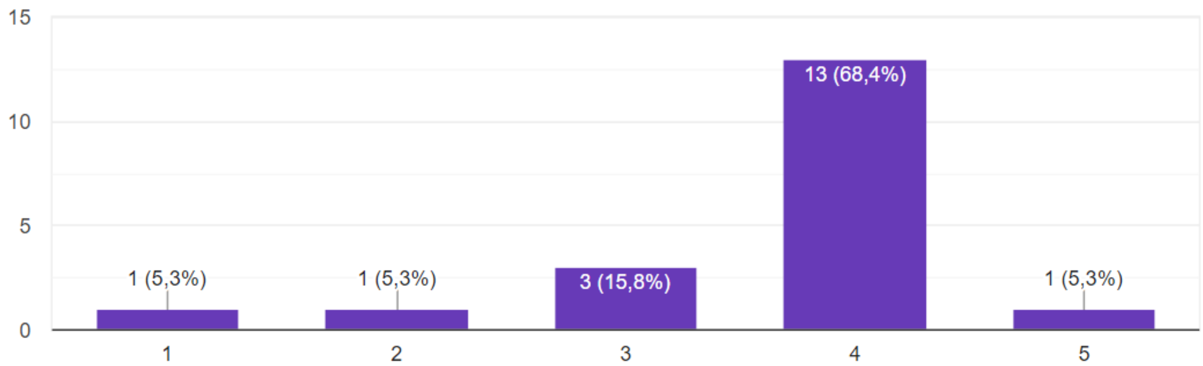
Question 15: are there uncertainties as to why the system has exactly detected stressful events?



Question 16 (optional): if so, which ones are these?

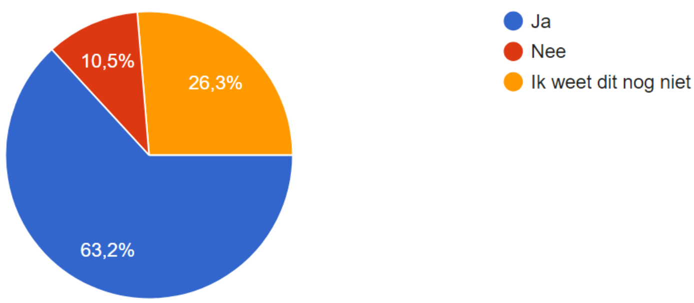
- Some participants do not recall seeing an exact explanation as to why stressful events were detected.
- It was unclear why on other similar moments, according to the metrics on the timeline, stress was not detected. It was also unclear what determined the length of the stress response.
- It is uncertain why the system only detects stress based on heart rate features. What happens if other parameters indicate stress but not the heart?
- Some peaks in metrics (eg. heart rate and breathing rate peak, HRV dip) don't seem significant during stress responses and are expected to be higher

Question 17: how well does the application accommodate to your personal needs or concerns regarding stress management and general health?



In this range, 1 equals to the application not accommodating to the participant's needs at all, whereas 5 equals to the application accommodating to the participant's needs very well. The results show that most participants would use the features in the application provides in order to track their health despite some of its current complications.

Question 18: would you use a similar stress management application in the future?



Question 19: if you already use an application for this, what's it called?

No participants seem to use a similar application, except for one. This participant doesn't remember the name, but had to use it when he felt his heart rate increase. It measured his heart rate, heart rhythm, and oxygen levels.

Question 20: do you have suggestions for extra functions or improvements which may make the application more useful or pleasant to use?

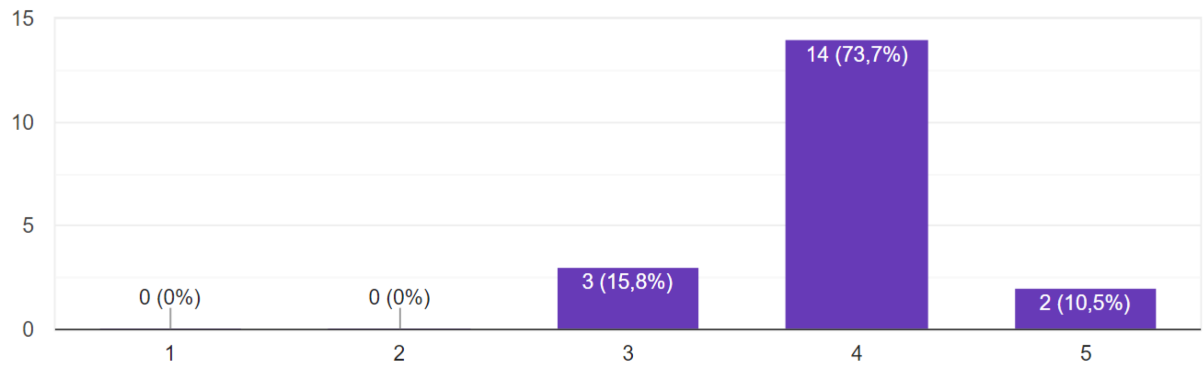
Improvements in user experience

- Add clear tooltips and overlays with icons and terms. Add explanations to parameters when hovering or clicking on them.
- Make some elements appear bigger. The moods in particular were the most important, but you can only see the icons clearly when zooming in a lot.
- Add a scroll bar to the timeline and an indicator for zooming.
- Consistent UI elements and better colors for icons.
- Add extra data to the legend which explains the relations between the data and their baselines, etc.

Addition of features

- Referral to help websites.
- An additional food logging system so the impact of diet can be linked to the metrics shown in the graphs.
- Add functionalities on a higher level. For example a feature which lets the user retrieve a list of moments in which he perceived stress, instead of letting him search all instances throughout the timeline manually.
- A mobile application to view the timeline and trends would be more useful.
- Add a Dutch version.

Question 21: On a scale from 1 to 5, how satisfied are you with this application in general ?



In this range, 1 equals to being totally unsatisfied with using the application for stress and health management, whereas 5 equals to being very satisfied. The vast majority of participants were satisfied with the application, proving its effectiveness for its purpose.

6.3 Discussion

6.3.1 Application contents

Despite the results from the studies being mostly positive, it is clear that there are some shortcomings that need to be addressed. The most obvious issue, judging from the recordings, is that the participants did not notice the help screen button (shown in figure at the top right). Before setting up the user studies, it was thought that the information in the help screen would be too much to put into tooltips. Yet, looking back at it later, this could be solved with a "more.." button or something similar. The issue resulted in many people not knowing the meaning of some of the metrics, particularly HRV and cEDA, although most participants could identify most of the stressors without them thanks to the logs. As mentioned by some of the participants, some sort of training would be needed to effectively derive more profound conclusions from the complicated metrics.

Another major aspect is the transparency behind the stress responses. Currently, when clicking on a stress response, deviations in heart rate features from normal features are shown to provide a potential explanation. Also, a rising trend in heart rate or a lowering trend in HRV could also explain why there was a stress response. It is noted that, because the kNN machine learning algorithm didn't work very well in the testing phase of the system, carefully logged stress moments were taken in order to "substitute" stress response data in the timeline. However, even with this substituted data, it remains unclear why stress was detected at a particular interval and not at another. During a stress response, it is possible that HRV and heart rate do not change significantly compared to an event where no stress was detected. The interaction between stress and the heart is a very complex phenomenon and cannot be accurately derived from heart rate and HRV alone. Maybe a health expert could derive patterns from the ECG signal during the stress response. Because it is impossible to plot all the raw data points in the timeline, an extra window could be implemented for a low-level analysis. Perhaps frequency-domain features and geometrical features could be shown in the timeline to make the stress responses more explainable, but there is a possibility that this would only further confuse the average user. On the positive side, there were no signs of 'distrust' from the participants about the detected stress. When a false positive occurs, the user may feel mildly annoyed at best. When a false negative occurs, the logged moods and activity serve as a backup. Having informed the user that the system can make errors, they can help improve the model's predictions by logging their moods for further training, and this is enough. The conclusion is that more explainability, besides notifying changes in the decisive features such as heart rate, is not as crucial in stress detection. The stakes of wrong predictions are relatively low compared to systems like autonomous driving.

6.3.2 Application performance

Despite not being content-related, another major thing that negatively impacted the study is the performance of the web application. This is unfortunately due to the choice of technology used to create the application. The first issue regarding the performance, was loading in the data. Each data point from the graphs in the timeline equalled to a html-element being placed on the canvas. Having a data point per second over a span of thirty minutes would therefore result in a lot of lagginess. To reduce this effect, some data points that do not contribute to visualizing the general trend are skipped in the back-end queries. Having minimized the amount of needed data points, the next step would be to limited the data points to the start and end date range of the timeline each time the user moves the timeline or zooms in, so only the necessary data is loaded instead of everything at once. To put it short, this gave a lot of synchronization errors between the different rows of the timeline because they need to delegate the zooming and dragging event to each other while updating its data. As a solution, only the data of the current day picked in the date picker was retrieved. However, this meant that dragging the timeline to the next day didn't result in the retrieval of data, as noticed by some participants. In order to fix this, all the timeline data which the user had to look for was loaded once into the timeline. While this data wasn't relatively much, it still took up to 15 seconds to load everything in. Especially the tooltips caused lagginess in the timeline, when hovering over the data points. Vis.js does not support tooltips and the values of the data points are not stored in the html; their positions relative to the size of the graph are. There is however an option to put labels next to the data points, containing the real y-values of the data. So, as a solution to provide tooltips, these label elements were visually hidden but detectable on hovering. This however led to many labels being loaded in, making the web page slower. When zoomed out, labels are stacked upon each other, so hovering would sometimes also hit the wrong label.

The development of the web application in Vis.js was a bit cumbersome overall, as the Timeline components needed a lot of modifications to make it look like the final result. Slight changes in layout resulted in strange behavior of the timeline, and adding labels left to the line graphs (which is not directly possible via options in Vis.js) had to be done manually, which often resulted in misalignment with other labels. One of the participants made a remark about no zoom percentage being shown. There was an attempt to create this feature. However, in Vis.js, you have a current zoom level and a max zoom level. When displaying the timeline on a day range, each zoom in causes the zoom level to increase tremendously, whereas zooming in from a minute range to a second range causes only a slight increase in this zoom level. This made scaling hard, so the priority was set on other timeline features. In the end, the result required a lot of code, even though it doesn't look like it at first sight, and all the struggling made me lose sight of some important UI/UX aspects. Maybe a library other than Vis.js could be used to create a timeline with better performance, or maybe a new type of visualization could be created in the future, as there are not really any timeline libraries supporting both 2D graphs and events without workarounds.

Chapter 7

Conclusions

7.1 Reflection

To create a system to measure and visualize parameters related to stress and well-being, it is important to do some background research about it. Starting off with the thesis, I had a strong feeling that this was related to the heart and I was interested in looking into it. This required reading and understanding medical papers, and often looking for the meaning of medical terms mentioned in those papers. Medical implications needed to be read thoroughly, since you can't just make arbitrary choices, especially when something is related to the medical field.There were moments of being stuck in the reading material, because there were so many medical aspects I had to take into account, so a concrete conclusion could not be formed from the literature study at first. When analysing the heart rate data acquired by the Polar H10 sensor, it was essential to look into the required preprocessing and feature extraction steps. This included learning new concepts about signal processing such as frequency analysis. There were also a lot of uncertainties about what to exactly do with the heart rate data after having it preprocessed. Eventually, the most important aspects were understood from the medical literature. All studies which propose solutions for detecting stress or disease went through more or less the same steps for computation: measure a relevant physiological signal, preprocess this signal, extract features from the signal, feed them to a machine learning algorithm and comparing different algorithms depending on the study. The moment I understood this, is was time to start focussing on the system to be built. In the case of this thesis, the machine learning algorithms were limited to using a kNN because it seemed the most intuitive at that time and suitable for the use case at the time of implementation: it is an algorithm which makes fast predictions, and new data for future predictions is added fast. However this algorithm did not perform well, which is due to it being inherently worse than other algorithms for stress detection, but also because there was not enough data. Because of this issue, only the web application of the system could be evaluated in the user study instead of the effectiveness of the system as a whole.

Before working on this thesis, I followed the courses User-centered software engineering and E-health during my first year in the Master's program. In the course User-centered software engineering, I learned a lot of interesting concepts in the theory of HCI. The final project in this course was developing a prototype application for High-Intensity Training against lower back pain. It was interesting to do something multidisciplinary-related, but the final prototype was never tested by its end users and only reviewed by people from outside the university who did not seem to care all that much. Therefore, it was rewarding to revisit this approach in my thesis, where I could truly test the user experience in my own way. During that time period, I also followed the course E-health in which I had to build a "smart toilet seat" with the help of sensors. The seat could identify people via Bluetooth and (attempted to) classify the visual properties of feces via a camera in order to inform elderly assistants. The teamwork did not go well however, and the end result could not even be tested practically. I am pleased that this thesis provided me with a second opportunity to develop a health-related system involving sensors, and that it gave me the freedom to create something I believed would be truly useful for the end user.

7.2 End result & improvements

To efficiently create a prototype, the data acquisition part of the system was separated into a mobile mobile application, a predictor and a data processor communicating over HTTP. Even though the system created for this thesis is just a prototype, HTTP communication can introduce some overhead. Normally, these things would all run in the same program on the same mobile device in a real final product. Because I did not have any experience with running machine learning algorithms on android devices while maintaining UI functionality and sensor connection, this was deliberately avoided. However, with some practice in task scheduling, it is definitely possible to perform live stress detection while updating a better performing model with new data. In the future, it would also definitely be interesting to try multiple machine learning algorithms and gather more data, possibly from different modalities. Regarding the web application, it was good to see that there was positive feedback on the design of the timeline from the user studies, and I am happy that I succeeded in developing part of my vision. It can be said that the parallel comparison of data definitely helped most of the users in identifying links between stress and other data, despite the shortcomings of the application.

Having researched related works, no application was found which allows for comparison of multiple data in the form of a timeline. Yet, one of the questions I asked myself after the studies, is whether the proposed application has any real benefit for the users compared to an existing application. The results of the user study are majorly positive, but there is no telling in how this translates to the current market. To have made the study more complete, it would be beneficial to let the users use an existing commercial app or develop an additional similar one, and perform a statistical analysis such as the t-test to see where one app excels and where it falls short compared to the other.

Chapter 8

Dutch summary

8.1 Probleem -en doelstelling

Stressniveaus, of ze nu voortkomen uit fysieke of mentale stress, spelen vaak een belangrijke rol in de context van welzijn. Wetenschappelijke studies hebben aangetoond dat langdurige werkgerelateerde stress een risicofactor is voor hart- en vaatziekten en bijdraagt aan sterfgevallen op de werkplek bij mensen met en zonder reeds bestaande cardiovasculaire aandoeningen. Het is daarom cruciaal om gegevens met betrekking tot gezondheid en andere lichaamsignalen die met deze aandoeningen verband houden grondig te onderzoeken, zonder daarbij fysiek een belemmering te vormen. Dit kan worden bereikt door het gebruik van draagbare sensoren zoals de hartslagmeter en de versnellingsmeter, die doorgaans zijn geïntegreerd in polsbandapparaten zoals smartwatches. Het is echter ook noodzakelijk om subjectieve stress te beoordelen om beter inzicht te krijgen in de exacte oorzaken van onderliggende gezondheidsfactoren die door de sensoren worden gemeten. Dit kunnen parameters zijn zoals waargenomen werkdruk en tevredenheid, of het type emotie dat op een bepaald moment wordt gevoeld. Daarom is een soort applicatie nodig waarmee de gebruiker deze parameters kan invoeren, bij voorkeur een mobielvriendelijke applicatie om efficiëntieredenen. Deze applicatie moet zowel de individuele gebruiker als iemand met medische expertise in staat stellen om zijn/haar fysieke en mentale gezondheidstoestand te monitoren. Het doel van deze proef is daarom om te onderzoeken hoe een dergelijk systeem kan worden ontwikkeld dat kan worden gebruikt om gezondheids- en stressgerelateerde fysiologische parameters te monitoren via draagbare sensoren zonder een hindering te vormen voor het individu. Daarnaast moet het systeem een mobielvriendelijke manier omvatten om subjectieve stressgegevens, zoals de waargenomen werkdruk, direct van de gebruiker te verzamelen. Om de monitoring van fysieke en mentale gezondheidsgegevens mogelijk te maken die door de mobiele applicatie worden verstrekt, wordt een dashboard ontwikkeld. Dit dashboard presenteert niet alleen ruwe gezondheidsgegevens, maar contextualiseert ook de gepresenteerde informatie door verbanden te illustreren tussen de gegevens en de door de gebruiker gegeven feedback. Het dashboard moet ook uitleg bieden over waarom de gebruiker op een bepaald moment stress ervoer. Deze eigenschap staat bekend als 'intelligibility', waardoor een duidelijker begrip van de gezondheids- en stresspatronen van de gebruiker mogelijk wordt.

8.2 Stressindicatoren

Om de link tussen stress en cardiovasculaire aandoeningen te begrijpen, is het nodig om enkele onderliggende biologische factoren te bespreken. Het autonome zenuwstelsel (ANS) speelt een cruciale rol in dit proces. Het ANS heeft twee primaire divisies: het sympathische zenuwstelsel (SNS) en het parasympathische zenuwstelsel (PNS). Het SNS wordt vaak geassocieerd met de "vecht-of-vluchtreactie", die het lichaam voorbereidt op fysieke activiteit en de hartslag verhoogt door cortisol in de bloedbaan af te geven. Het PNS bevordert spijsvertering, ontspanning en energiebesparing, wat leidt tot een verlaging van de hartslag. Het ANS beïnvloedt de activiteit van de sinusknoop, de natuurlijke pacemaker van het hart, en is vatbaar voor de slijtage-effecten van allostatische belasting. Dit kan het vermogen om zich aan stress aan te passen in gedrang brengen, waardoor het risico op de ontwikkeling van hart- en vaatziekten toeneemt.

Hartslag (HR) en hartslagvariabiliteit (HRV) zijn belangrijke indicatoren van de functie van het ANS. HR wordt gemeten met behulp van een elektrocardiogram (ECG), dat elektrische signalen detecteert die door het hart worden geproduceerd. HR is het aantal R-golven dat in een minuut wordt geregistreerd, uitgedrukt in slagen per minuut (bpm). Het analyseren van HR tijdens rust is essentieel, aangezien er een significante correlatie is tussen rustende HR en het risico op hart- en vaatziekten en plotselinge hartdood. HRV meet de tijdsintervallen tussen opeenvolgende hartslagen (RR-intervallen) en geeft inzicht in de functie van het ANS. Hoge HRV duidt op een gezond, responsief ANS, terwijl verminderde HRV wordt gekoppeld aan stress en angst, wat duidt op verminderde parasympathische activiteit. Er zijn verschillende methoden om HRV te beoordelen. Tijdsdomeinanalyse onderzoekt veranderingen in het ECG-signaal in de tijd, waarbij de wortel van de gemiddelde kwadraatafwijkingen van opeenvolgende verschillen (RMSSD) een veelgebruikte methode is om parasympathische activiteit te beoordelen. Frequentiedomeinanalyse verkent de verschillende frequentiebanden in RR-intervallen. De hoogfrequente (HF) band, variërend van 0,15 tot 0,4 Hz, duidt op parasympathische activiteit en correleert met de ademhalingscyclus. De laagfrequente (LF) band, variërend van 0,04 tot 0,15 Hz, wordt beïnvloed door zowel sympathische als parasympathische activiteit en wordt geassocieerd met baroreceptoractiviteit, wat helpt bij het handhaven van bloeddrukhomeostase. Niet-lineaire methoden, zoals de Poincaré-plot, visualiseren HRV met behulp van een scatterplot van RR-intervallen, wat een geometrische analyse van de balans tussen het SNS en PNS biedt.

Ademhaling en huidgeleiding zijn aanvullende stressmarkers. Respiratoire sinusaritmie (RSA) koppelt ademhalingspatronen aan hartslagveranderingen. Gecontroleerd, langzaam ademhalen verbetert cardiovasculaire en cognitieve functies en helpt bij het beheersen van stress en angst. Elektrodermale activiteit (EDA) meet de elektrische geleidbaarheid van de huid, wat de zweetproductie en de activiteit van het sympathische zenuwstelsel weerspiegelt.

Voor nauwkeurige HRV-analyse is het belangrijk om artefacten en ectopische slagen uit ECG-metingen te filteren. Artefacten, zoals veranderingen in het elektrische signaal die niet door cardiale activiteit worden veroorzaakt, kunnen worden gedetecteerd en uitgefilterd met behulp van technieken zoals lineaire interpolatie. Ectopische slagen, onregelmatigheden in het hartritme, moeten ook worden uitgesloten van HRV-berekeningen, omdat ze de ANS-activiteit niet weerspiegelen. Een lage HRV kan wijzen op problemen met het ANS, maar het is cruciaal om bijkomende medische aandoeningen en individuele verschillen in overweging te nemen voordat conclusies worden getrokken op basis van uitsluitend HR- en HRV-metingen.

8.2.1 Meten en verzamelen data

De conventionele methode om de hartslag te meten met behulp van ECG-monitors in klinische omgevingen vormt een belemmering door onder andere de losse kabels. Als reactie op deze beperking komt de Polar H10 hartmonitor naar voren als oplossing. Deze borstband is uitgerust met een enkelvoudige elektrode, die nauwkeurige metingen levert die vergelijkbaar zijn met die verkregen van ECG-monitors in een ziekenhuis. De borstband, bekend als de Polar ProStrap, is specifiek ontworpen om signaalruis te minimaliseren, wat zorgt voor nauwkeurige hartslagmetingen zelfs tijdens intense fysieke activiteit.

De Polar H10 maakt gebruik van een real-time signaaldetectie-algoritme om QRS-complexen nauwkeurig te identificeren uit het ECG-signaal met sub-milliseconde precisie, wat bijdraagt aan de betrouwbaarheid bij het meten van de hartslag. Gegevens verzameld door de Polar H10 kunnen draadloos worden verzonden via Bluetooth. Deze mogelijkheid tot gegevensoverdracht maakt naadloze integratie mogelijk met toepassingen zoals de Polar Beat-app, waardoor gebruikers fysieke activiteitssessies kunnen starten en volgen. Bovendien is de Polar H10 uitgerust met een tri-axis versnellingsmeter voor bewegingsdetectiedoeleinden, hoewel deze functie niet diepgaand wordt onderzocht in deze proef.

Het verwerken van de gegevens verkregen van de Polar H10 omvat verschillende stappen die worden vergemakkelijkt door Python-libraries zoals pandas, matplotlib, HRVanalysis, pyHRV en heartpy. Deze libraries maken het opschonen van signalen, verwijdering van ectopische slagen en berekening van hartslagvariabiliteit (HRV) mogelijk. Om basale gegevens vast te stellen voor individuele analyse, worden opnames gemaakt tijdens verschillende dagelijkse activiteiten, als referentie voor vergelijking tijdens stressvolle gebeurtenissen. Deze opnames leggen fysiologische reacties vast tijdens zowel fysieke inspanning als mentaal stressvolle situaties, wat inzicht biedt in hoe hartslag en HRV fluctueren onder verschillende omstandigheden.

8.3 Stress management systeem

Een stressmanagement-prototypesysteem is ontworpen om de gezondheid van individuen efficiënt en onopvallend te monitoren. Het overkoepelende doel is gebruikers te voorzien van verschillende functies om hun mentale en fysieke gezondheidsparameters effectief te meten en te beheren, waarbij zowel fysiologische gegevens als subjectieve feedback worden benut. Het systeem bestaat uit vier componenten: de mobiele applicatie, de gegevensverwerker, de stressvoorspeller en de webapplicatie.

8.3.1 Mobiele applicatie

De mobiele applicatie is gebouwd met behulp van Kotlin en de Polar SDK, specifiek ontworpen voor Android- en iOS-applicaties om te communiceren met Polar-sensoren zoals de H10. Het is gebaseerd op een startproject dat wordt voorzien in de Polar GitHub-repository, dat functies bevat voor communicatie met de H10-sensor. De hoofdfunctionaliteit van de applicatie draait om het verbinden met de H10 sensor via Bluetooth, gegevens streamen en het loggen van activiteiten en stemmingen. Gebruikers starten verbindingen door de apparaatidentificatie te verstrekken die op de sensor wordt weergegeven. Eenmaal verbonden, kunnen ze activiteiten starten of hun stemming loggen. Na het voltooien van een activiteit vraagt de app de gebruiker om hun stemming te loggen, met als doel een machine learning-algoritme te trainen op de verzamelde hartslaggegevens gelabeld per stemming. Daarnaast luistert de app naar stressberichten van de dataprocessor-API. Bij het detecteren van een stressvolle gebeurtenis genereert het een pushmelding om de gebruiker te informeren. Door op de melding te klikken wordt de gebruiker gevraagd om opnieuw hun stemming te loggen, wat bijdraagt aan de accuraatheid van het machine learning-model.

Om gelijktijdige operaties uit te voeren zoals luisteren naar stressberichten, RR-intervallen naar de API sturen, activiteiten en stemmingen loggen en reageren op UI-gebeurtenissen te verwerken, maakt de applicatie gebruik van meerdere threads. De hoofdthread beheert UI-updates en sensorgegevensstreaming, terwijl afzonderlijke threads netwerkoperaties en achtergrondtaken afhandelen. Een producent-consumentparadigma vergemakkelijkt de communicatie tussen threads en voorkomt crashes.

8.3.2 Gegevensverwerker

De data processor API dient als de "backbone" van het systeem, en is verantwoordelijk voor het verwerken van binnenkomende gegevens, het omzetten van RR-intervallen in hartslagfeatures, het doorgeven van deze features aan de stressvoorspeller en het opslaan ervan in de database voor visualisatiedoeleinden. De gegevensverwerker is volledig geschreven in Python, maakt het gebruik van de FastAPI-library om HTTP-verzoeken van de web- en mobiele applicaties te verwerken. FastAPI's lichte karakter en eenvoudige opzet maakt het ideaal voor snelle prototyping en eenvoudigere use cases in vergelijking met frameworks zoals Java Spring Boot of C# ASP.NET. Het integreren van functionaliteit voor gegevensverwerking in de mobiele app zou complexiteit toevoegen, en het gebrek aan bekendheid met machine learning-libraries in Java leidde tot de oplossing van de FastAPI-backend in Python. Hoewel Python de implementatie van algoritmen en de load balancing vereenvoudigt, vereist het wel een netwerkverbinding.

In de gegevensverwerker worden RR-intervallen gebufferd, waaruit hartslagkenmerken elke minuut worden berekend. Deze kenmerken, samen met hartslag, HRV en ademhalingsfrequentie, worden rechtstreeks opgeslagen in PostgreSQL-tabellen. De RR-intervallen zelf worden ook opgeslagen in een aparte tabel. Wanneer een gebruiker zijn stemming in de app registreert, worden RR-intervallen van de voorafgaande drie minuten opgehaald voor het berekenen van de hartslagkenmerken die worden gebruikt bij het trainen van het machine learning-model.

8.3.3 Stressvoorspeller

De stressvoorspeller in de mobiele applicatie maakt gebruik van het k-nearest neighbors (kNN) algoritme voor classificatie. Het model traint continu naarmate er nieuwe stemminggegevens worden vastgelegd in de mobiele applicatie, wat in de praktijk wordt genoemd als 'online learning'. De initiële trainingsgegevens zijn afkomstig van de ECG subset van de WESAD dataset, voornamelijk verzameld bij jonge, gezonde studenten. De voorspeller houdt rekening met 18 HRV features voor training, waaronder gemiddelde RR-interval, standaardafwijking van RR-intervallen, hartslag en spectrale dichtheid van RR-intervallen in verschillende frequentiebanden. Ondanks het behalen van een hoge nauwkeurigheid tijdens het testen, heeft het model moeite met generalisatie wanneer het wordt toegepast op ongeziene gegevens als gevolg van een gebrek aan variatie in de trainingsdataset.

Uiteindelijk is de prestatie van het kNN model suboptimaal in vergelijking met andere machine learning technieken, mede door het beperkte variatie in de trainingsdata. Desalniettemin is het systeem functioneel in staat om voorspellingen te maken (ongeacht de correctheid van deze voorspellingen), en traint het continu het model met nieuwe stemminggegevens zoals verwacht.

8.3.4 Webapplicatie

De webapplicatie, gebouwd met JavaScript en ReactJS, dient als instrument voor gebruikers om in hun fysieke en mentale gezondheidsgegevens te duiken. De op componenten gebaseerde architectuur van ReactJS maakt flexibele en schaalbare gebruikersinterfaces mogelijk, terwijl Vis.js de visualisatiemogelijkheden voorziet, met name bij het presenteren van tijdlijnen en lijngrafieken.

De tijdlijnfunctionaliteit van de applicatie biedt een interactief platform voor gebruikers om hun gegevens gedetailleerd te verkennen. Door in te zoomen, te slepen en specifieke tijdsintervallen te selecteren, kunnen gebruikers trends en patronen in hun fysiologische metingen, stemminglogs en stressvoorspellingen analyseren. De opname van emoticons en kleurgecodeerde representaties voegt een laag van intuïtiviteit toe, waardoor het voor gebruikers eenvoudiger wordt om hun gegevens te interpreteren. Verder biedt de trends-pagina een uitgebreid overzicht van gezondheids- en stressbeheermetingen in de loop van de tijd. Gebruikers kunnen hun voortgang bijhouden via visualisaties van gelogde stemmingen, fysiologische metingen, stressreacties en niveaus van lichamelijke activiteit. Correlaties tussen verschillende metingen kunnen worden afgeleid, waardoor waardevolle inzichten ontstaan in hoe levensstijlfactoren de algehele gezondheid kunnen beïnvloeden.

Ondanks mogelijke uitdagingen met bijvoorbeeld onvolledige sensorgegevens tijdens inactieve periodes, handhaaft de applicatie transparantie door opgenomen datapunten weer te geven en uitleg te bieden via tooltips. De opname van medische informatie en bronnen zorgt ervoor dat gebruikers toegang hebben tot relevante context bij het interpreteren van hun gegevens.

8.4 Gebruikerstudie

Na het ontwikkelen van een prototype systeem voor stress- en gezondheidsmanagement, is het essentieel om te meten hoe effectief het is wanneer anderen het gebruiken. In deze proef ligt de focus volledig op het bestuderen van het gebruik van de webapplicatie. Het evalueren van de effectiviteit van het stressdetectie-algoritme tijdens het dragen van de Polar hartslagsensor en het gebruik van de mobiele applicatie wordt bemoeilijkt door de grote hoeveelheid valse positieven in gedetecteerde stressreacties en andere beperkingen van middelen. Daarom is het doel van dit onderzoek om de bruikbaarheidsaspecten van de webapplicatie te beoordelen en de begrijpelijkheid van de gepresenteerde gegevens in de visualisaties te evalueren, waarbij tevredenheid wordt gekwantificeerd aan de hand van zowel scores als aanvullende tekstuele feedback. Na het analyseren van de resultaten van de studie kan worden bepaald hoe nuttig de tijdlijnbenadering was in het contextualiseren van fysiologische signalen met subjectieve feedback.

8.4.1 Verwerving participanten

Deelnemers moeten minstens 17 jaar oud zijn om te voldoen aan de wettelijke voorschriften met betrekking tot gegevensprivacy en toestemming. Geen andere specifieke criteria zijn vereist om deel te nemen, aangezien het monitoren van stress en gezondheid een activiteit is die openstaat voor de algemene bevolking. In totaal worden 19 deelnemers, die persoonlijk bekend zijn, geworven via directe berichten via platforms zoals Facebook en Discord, evenals via persoonlijke interactie.

8.4.2 Methode & procedure

Tijdens de tests worden de acties van de gebruiker opgenomen. Dit gebeurt om het gebruik te timen en eventuele fouten van de deelnemer te beoordelen om te voorkomen dat ze over het hoofd worden gezien door de observator. Het principe van hardop denken wordt gebruikt, wat betekent dat de gebruiker continu moet praten over de acties die ze uitvoeren. Met dit principe wordt hun vocale redenering opgenomen tijdens het proces van het gebruik van de webapplicatie. Dit kan helpen bij het beoordelen van hoe duidelijk de gebruikersinterface is en in welke aspecten het ontbreekt.

De studie zelf vindt plaats zowel in de HCI-demozaal van de EDM als bij de deelnemers thuis in een stille ruimte. Om eventuele storingen te voorkomen, hangt er een bordje aan de deur buiten. De deelnemer wordt verwelkomd door de supervisor en wordt gevraagd plaats te nemen voor een voorbereide laptop aan de tafel. Aanvankelijk wordt de deelnemer gevraagd om een blad te lezen met het doel van de studie en hoe deze zal worden uitgevoerd, en vervolgens een toestemmingsformulier te lezen en te ondertekenen. Als de deelnemer weigert, worden ze bedankt en naar buiten geleid. Als de deelnemer akkoord gaat om te tekenen, krijgen ze een blad met informatie die nodig is om de webapplicatie te evalueren. Dit blad bevat de beschrijving van een persoon die hypothetisch gebruik heeft gemaakt van de mobiele applicatie om fysiologische gegevens te registreren en feedback te geven via gelogde stemmingen en activiteiten. Het beschrijft twee soorten situaties waarin deze persoon bepaalde stressvolle gebeurtenissen wil terughalen door de webapplicatie te gebruiken om de oorzaken en correlaties van de ervaren stress te bepalen. Bovendien beschrijft het ook een situatie waarin deze persoon wil bepalen hoe zijn algehele gezondheid zich heeft ontwikkeld. Na het lezen van het informatieve blad wordt de deelnemer nogmaals herinnerd aan het principe van hardop denken, en wordt de opname gestart. Tijdens het gebruik van de webapplicatie mag de gebruiker het informatieve blad raadplegen. Als de gebruiker lang vastzit of klachten heeft, zal de waarnemer ingrijpen. Anders zal de waarnemer op afstand blijven om de deelnemer niet zenuwachtig te maken. Na het voltooien van alle situaties wordt de opname gestopt en wordt de gebruiker gevraagd om een online vragenlijst in te vullen.

8.4.3 Discussie

Ondanks over het algemeen positieve resultaten uit de studies, zijn er opmerkelijke gebieden voor verbetering. Ten eerste merkten deelnemers vaak de knop voor het helpscherm met meer medische informatie niet op, wat leidde tot een gebrek aan begrip van bepaalde metrieken zoals HRV en cEDA. Bovendien zijn er transparantieproblemen met stress responses, omdat het nog steeds onduidelijk is waarom stress op bepaalde momenten wordt gedetecteerd door het systeem. De interactie tussen stress en hartmetrieken is complex en vereist mogelijk expertiseanalyse buiten alleen hartslag en HRV. Deelnemers uitten echter over het algemeen geen wantrouwen in het stressdetectiesysteem, wat suggereert dat meer verklaringsmogelijkheden niet cruciaal zijn.

Wat betreft de prestaties van de applicatie waren er significante problemen met betrekking tot de keuze van technologie, met name met betrekking tot het laden van gegevens en synchronisatiefouten. Het laden van gegevenspunten veroorzaakte vertraging, vooral met tooltips, wat de gebruikerservaring beïnvloedde. Vis.js, de gekozen bibliotheek voor visualisatie, presenteerde uitdagingen bij het aanpassen van de tijdlijncomponenten om het gewenste ontwerp te bereiken. Problemen zoals het niet goed uitlijnen van labels en het ontbreken van feedback over het zoompercentage maakten de gebruikersinterface verder gecompliceerd. Hoewel er pogingen zijn ondernomen om deze problemen aan te pakken, benadrukken ze de noodzaak van alternatieve bibliotheken of nieuwe visualisatiebenaderingen voor verbeterde prestaties en gebruikerservaring in toekomstige iteraties van de applicatie.

8.5 Conclusies

8.5.1 Reflectie

Om een systeem te creëren dat parameters met betrekking tot stress en welzijn meet en visualiseert, is het belangrijk om eerst wat achtergrondonderzoek te doen. Bij het begin van mijn proef had ik het sterke gevoel dat dit gerelateerd was aan het hart, en ik was geïnteresseerd om hier verder in te duiken. Dit vereiste het lezen en begrijpen van medische artikelen, waarbij ik vaak moest zoeken naar de betekenis van medische termen die in die artikelen werden genoemd. De medische implicaties moesten grondig worden gelezen, omdat je niet zomaar willekeurige keuzes kunt maken, vooral niet als iets met het medische veld te maken heeft. Er waren momenten waarop ik vastzat in het leesmateriaal, omdat er zoveel medische aspecten waren waarmee ik rekening moest houden, waardoor er aanvankelijk geen concrete conclusie uit de literatuurstudie kon worden getrokken.

Bij het analyseren van de hartslaggegevens die door de Polar H10-sensor zijn verkregen, was het essentieel om te kijken naar de vereiste preprocessing- en feature-extractiestappen. Dit omvatte het leren van nieuwe concepten over signaalverwerking, zoals frequentieanalyse. Er waren ook veel onzekerheden over wat precies met de hartslaggegevens te doen na de preprocessing. Uiteindelijk werden de belangrijkste aspecten uit de medische literatuur begrepen. Alle studies die oplossingen voorstellen voor het detecteren van stress of ziekte doorlopen min of meer dezelfde stappen voor de berekening: een relevant fysiologisch signaal meten, dit signaal preprocessen, features uit het signaal extraheren, deze invoeren in een machine learning-algoritme en verschillende algoritmen vergelijken afhankelijk van de studie. Toen ik dit eenmaal begreep, was het tijd om me te concentreren op het te bouwen systeem. In het geval van deze proef waren de machine learning-algoritmen beperkt tot het gebruik van een kNN, omdat dit destijds het meest intuïtieve en geschikte algoritme leek voor de use case: het is een algoritme dat snelle voorspellingen doet en nieuwe gegevens voor toekomstige voorspellingen snel toevoegt. Dit algoritme presteerde echter niet goed, wat te wijten is aan het feit dat het inherent slechter is dan andere algoritmen voor stressdetectie, maar ook omdat er niet genoeg gegevens waren. Vanwege dit probleem kon alleen de webapplicatie van het systeem worden geëvalueerd in de gebruikersstudie in plaats van de effectiviteit van het systeem als geheel.

Voor het werken aan deze proef volgde ik in mijn eerste jaar van de masteropleiding de vakken User-centered software engineering en E-health. In de cursus Gebruikersgerichte software engineering leerde ik veel interessante concepten in de theorie van HCI. Het eindproject in deze cursus was het ontwikkelen van een prototype-applicatie voor High-Intensity Training tegen lage rugpijn. Het was interessant om iets multidisciplinairs te doen, maar het eindprototype werd nooit getest door de eindgebruikers en werd alleen beoordeeld door mensen van buiten de universiteit die niet erg betrokken leken. Daarom was het lonend om deze aanpak opnieuw te bekijken in mijn proef, waar ik de gebruikerservaring echt op mijn eigen manier kon testen. In die periode volgde ik ook de cursus E-health waarin ik een "slimme toiletbril" moest bouwen met behulp van sensoren. De bril kon mensen identificeren via Bluetooth en (probeerde) de visuele eigenschappen van ontlasting te classificeren via een camera om ouderenverzorgers te informeren. Het teamwork verliep echter niet goed en het eindresultaat kon zelfs niet praktisch worden getest. Ik ben blij dat deze proef me een tweede kans bood om een gezondheidsgerelateerd systeem met sensoren te ontwikkelen, en dat het me de vrijheid gaf om iets te creëren waarvan ik geloofde dat het echt nuttig zou zijn voor de eindgebruiker.

8.5.2 Eindresultaat & verbeteringen

Om efficiënt een prototype te creëren, werd het data-acquisitiegedeelte van het systeem gescheiden in een mobiele applicatie, een voorspeller en een dataverwerker die communiceren via HTTP. Hoewel het systeem dat voor deze proef is gemaakt slechts een prototype is, kan HTTP-communicatie enige overhead met zich meebrengen. Normaal gesproken zouden deze dingen allemaal in hetzelfde programma op hetzelfde mobiele apparaat draaien in een echt eindproduct. Omdat ik geen ervaring had met het uitvoeren van machine learning-algoritmen op Android-apparaten terwijl ik de UI-functionaliteit en sensorverbindingen handhaafde, werd dit bewust vermeden. Met enige ervaring in thread management is het echter zeker mogelijk om live stressdetectie uit te voeren terwijl een beter presterend model met nieuwe gegevens wordt bijgewerkt. In de toekomst zou het ook zeker interessant zijn om meerdere machine learning-algoritmen te proberen en meer gegevens te verzamelen, mogelijk uit verschillende modaliteiten. Wat betreft de webapplicatie was het goed om te zien dat er positieve feedback was over het ontwerp van de tijdlijn uit de gebruikersstudies, en ik ben blij dat ik erin geslaagd ben een deel van mijn visie te ontwikkelen. Er kan worden gezegd dat de parallelle vergelijking van gegevens de meeste gebruikers zeker heeft geholpen bij het identificeren van verbanden tussen stress en andere gegevens, ondanks de tekortkomingen van de applicatie.

Tijdens het onderzoek naar gerelateerde werken werd er geen toepassing gevonden die vergelijking van meerdere gegevens in de vorm van een tijdlijn mogelijk maakt. Toch stelde ik mezelf na de studies de vraag of de voorgestelde applicatie echt voordeel biedt voor de gebruikers vergeleken met een bestaande applicatie. De resultaten van de gebruikersstudie zijn overwegend positief, maar het is niet te zeggen hoe dit zich vertaalt naar de huidige markt. Om de studie completer te maken, zou het nuttig zijn geweest om de gebruikers een bestaande commerciële app te laten gebruiken of een extra vergelijkbare te ontwikkelen, en een statistische analyse zoals de t-test uit te voeren om te zien waar de ene app uitblinkt en waar deze tekortschiet in vergelijking met de andere.

Bibliography

[1] Study-boosting benefits of exercise. https://www.ucl.ac.uk/students/news/2020/may/study-boosting-benefits-exercise/, 05 2020. Accessed: 2024-06-10.

[2] Biostrap. Biostrap mobile app. https://biostrap.com/wp-content/uploads/2023/07/smartphone.png, 2023. Accessed: 2024-06-10.

[3] BrightestXC. Pixel watch reviews. https://www.reddit.com/r/PixelWatch/comments/17fen7h/body_response_notifications/. Accessed: 2024-06-10.

[4] McEwen BS. Stress, adaptation, and disease. allostasis and allostatic load. *Annals of the New York Academy of Sciences*, 05 1998.

[5] Robin Champseix. Aura github. https://github.com/Aura-healthcare/hrv-analysis?tab=readme-ov-file. Accessed: 2024-06-10.

[6] Theodora Chaspari. Example of an electrodermal activity (eda) signal of skin conductance responses (scr), marked with red "o," and an indicative notation of scr amplitude measure. https://www.researchgate.net/figure/Example-of-an-electrodermal-activity-EDA-signal-of-skin-conductance-responses-SCR_fig1_269416229, 12 2014. Accessed: 2024-06-10.

[7] Theodora Chaspari, Andreas Tsiartas, Leah I Stein, Sharon A Cermak, and Shrikanth S Narayanan. Sparse representation of electrodermal activity with knowledge-driven dictionaries. *IEEE Trans Biomed Eng*, 62(3):960–971, 12 2014.

[8] Jerry Chen, Maysam Abbod, and Jiann-Shing Shieh. Pain and stress detection using wearable sensors and devices—a review. *Sensors*, 21:1030, 02 2021.

[9] Europese Commissie. Mogen de persoonsgegevens van kinderen worden verzameld? https://commission.europa.eu/law/law-topic/data-protection/reform/rights-citizens/how-my-personal-data-protected/can-personal-data-about-children-be-collected_nl#voorbeelden. Accessed: 2024-06-10.

[10] David Cornforth, Mika Tarvainen, and Herbert Jelinek. How to calculate renyi entropy from heart rate variability, and why it matters for detecting cardiac autonomic neuropathy, 09 2014.

[11] Padraig Cunningham and Sarah Delany. k-nearest neighbour classifiers. *Mult Classif Syst*, 54, 04 2007.

[12] Samy Abdel-Ghaffar Elena Perez. How we trained fitbit's body response feature to detect stress. https://blog.google/products/fitbit/how-we-trained-fitbits-body-response-feature-to-detect-stress/, 2023. Accessed: 2024-06-10.

[13] Fitbit. Google pixel watch 2. https://www.fitbit.com/global/en-ca/products/smartwatches/google-pixel-watch2. Accessed: 2024-06-10.

[14] Giorgos Giannakakis, Kostas Marias, and Manolis Tsiknakis. A stress recognition system using hrv parameters and machine learning techniques. In *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pages 269–272, 2019.

[15] Willem Gielen, Kevin Andrew Longoria, and Reinier Alexander van Mourik. Two cases of COVID-19 monitored by a wearable biosensor-a case report. *Mhealth*, 7:62, 10 2021.

[16] Google. Pixel watch 2. https://store.google.com/us/product/pixel_watch_2_specs?hl=en-US. Accessed: 2024-06-10.

[17] Murat Kaçar, Irina Savelieva, Harry Hemingway, and Marek Malik. Automatic ectopic beat elimination in short-term heart rate variability measurement. *Computer methods and programs in biomedicine*, 63:123–31, 11 2000.

[18] Mika Kivimäki, Jaana Pentti, Jane Ferrie, G Batty, Solja Nyberg, Markus Jokela, Marianna Virtanen, Lars Alfredsson, Nico Dragano, Eleonor Fransson, Marcel Goldberg, Anders Knutsson, Markku Koskenvuo, Aki Koskinen, Anne Kouvonen, Ritva Luukkonen, Tuula Oksanen, Reiner Rugulies, Johannes Siegrist, and John Deanfield. Work stress and risk of death in men and women with and without cardiometabolic disease: a multicohort study. *The Lancet Diabetes Endocrinology*, 6, 06 2018.

[19] Marcin Kołodziej. A new method of cardiac sympathetic index estimation using a 1d-convolutional neural network. *Bulletin of the Polish Academy of Sciences Technical Sciences*, 69, 06 2021.

[20] Marc K. Lahiri, Prince J. Kannankeril, and Jeffrey J. Goldberger. Assessment of autonomic function in cardiovascular disease: physiological basis and prognostic implications. *Journal of the American College of Cardiology*, 05 2008.

[21] Adrian Low and Rollin McCraty. Heart rate variability: New perspectives on assessment of stress and health risk at the workplace. *Heart and Mind*, 2:16, 01 2018.

[22] Emily A. Lundstrom, Mary Jane De Souza, Kristen J. Koltun, Nicole C.A. Strock, Hannah N. Canil, and Nancy I. Williams. Wearable technology metrics are associated with energy deficiency and psychological stress in elite swimmers. *International Journal of Sports Science & Coaching*, 10 2023.

[23] J. McNames, C. Crespo, M. Aboy, J. Bassale, L. Jenkins, and B. Goldstein. Harmonic spectrogram for the analysis of semi-periodic physiologic signals. In *Proceedings of the Second Joint 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society] [Engineering in Medicine and Biology*, volume 1, pages 143–144 vol.1, 2002.

[24] Giovanni Minarini. Root mean square of the successive differences as marker of the parasympathetic system and difference in the outcome after ans stimulation. 02 2020.

[25] Eric Morgan. All about hrv part 4: Respiratory sinus arrhythmia. `https://support.mindwaretech.com/2017/09/all-about-hrv-part-4-respiratory-sinus-arrhythmia/`, 09 2017. Accessed: 2024-06-10.

[26] J E Muller, G H Tofler, and R L Verrier. Sympathetic activity as the cause of the morning increase in cardiac events. a likely culprit, but the evidence remains circumstantial. *Circulation*, 91(10):2508–2509, 05 1995.

[27] Tobias Opthof and Ruben Coronel. The normal range and determinants of the intrinsic heart rate in man. *Cardiovascular research*, 45 1:177–84, 01 2000.

[28] Polar. Polar whitepapers. `https://www.polar.com/en/science/whitepapers`. Accessed: 2024-06-10.

[29] ReactiveX. Reactivex. `https://reactivex.io/intro.html`. Accessed: 2024-06-10.

[30] Rebecca Robbins. Why you wake up at 3am — and how to go back to sleep - rebecca robbins. `https://ouraring.com/blog/why-do-i-wake-up-at-3-am/`. Accessed: 2024-06-10.

[31] Marcelle Schaffarczyk, Bruce Rogers, Rüdiger Reer, and Thomas Gronwald. Validity of the polar h10 sensor for heart rate variability analysis during resting state and incremental exercise in recreational men and women. *Sensors*, 22, 08 2022.

[32] Caroline Sevoz-Couche and Sylvain Laborde. Heart rate variability and slow-paced breathing:when coherence meets resonance. *Neuroscience Biobehavioral Reviews*, 135:104576, 02 2022.

[33] AG Shaper, G Wannamethee, PW Macfarlane, and M Walker. Heart rate, ischaemic heart disease, and sudden cardiac death in middle-aged british men. *British heart journal*, 70(1):49—55, July 1993.

[34] Márcio Dahia Silvan Ferreira, Esdras Costa. Visualization of k-nearest neighbors with two classes (blue circles and red triangles) with k = 3 and k = 7. `https://www.researchgate.net/figure/Visualization-of-k-Nearest-Neighbors-with-two-classes-blue-circles-and-red-triangles_fig3_359786522`, 04 2022. Accessed: 2024-06-10.

[35] Elena Smets, Emmanuel Rios Velazquez, Giuseppina Schiavone, Imen Chakroun, Ellie D'Hondt, Walter De Raedt, Jan Cornelis, Olivier Janssens, Sofie Van Hoecke, Stephan Claes, Ilse Van Diest, and Chris Van Hoof. Large-scale wearable data reveal digital phenotypes for daily-life stress detection. *NPJ Digit Med*, 1:67, 11 2018.

[36] Statbel. Doodsoorzaken. https://statbel.fgov.be/nl/themas/bevolking/sterfte-en-levensverwachting/doodsoorzaken. Accessed: 2024-06-10.

[37] Jian Sun. Research on vocal sounding based on spectrum image analysis. *EURASIP Journal on Image and Video Processing*, 01 2019.

[38] Mazhar B. Tayel and Eslam I. AlSaba. Poincaré plot for heart rate variability. *International Journal of Biomedical and Biological Engineering*, 9(9), 01 2015.

[39] TeemuLauPolar. Polar sdk github. https://github.com/polarofficial/polar-ble-sdk. Accessed: 2024-06-10.

[40] Eleonora Tobaldini, Lino Nobili, Silvia Strada, Karina R Casali, Alberto Braghiroli, and Nicola Montano. Heart rate variability in normal and pathological sleep. *Front Physiol*, 4:294, 10 2013.

[41] Toppr. Linear interpolation formula. https://www.toppr.com/guides/maths-formulas/linear-interpolation-formula/. Accessed: 2024-06-10.

[42] Sebastian Uijtdehaage and Julian Thayer. Accentuated antagonism in the control of the human heart. *Clinical Autonomic Research*, 10:107–110, 06 2000.

[43] Ken Umetani, Donald Singer, Rollin McCraty, and Mike Atkinson. Twenty-four hour time domain heart rate variability and heart rate: Relations to age and gender over nine decades. *Journal of the American College of Cardiology*, 31:593–601, 03 1998.

[44] Wikipedia. Baroreceptor. https://en.wikipedia.org/wiki/Baroreceptor. Accessed: 2024-06-10.

[45] Wikipedia. Discrete fourier transformation. https://en.wikipedia.org/wiki/Discrete_Fourier_transform. Accessed: 2024-06-10.

[46] Wikipedia. Fast fourier transform. https://en.wikipedia.org/wiki/Fast_Fourier_transform. Accessed: 2024-06-10.

[47] Wikipedia. Fourier transform - wikipedia. https://en.wikipedia.org/wiki/Fourier_transform. Accessed: 2024-06-10.

[48] Wikipedia. Power spectral density. https://en.wikipedia.org/wiki/Spectral_density#:~:text=The%20power%20spectral%20density%20(PSD)%20of%20the%20signal%20describes%20the,hertz%20(W%2FHz). Accessed: 2024-06-10.

[49] Wikipedia. Welch's method. https://en.wikipedia.org/wiki/Welch%27s_method. Accessed: 2024-06-10.

[50] Lina Zhao, Peng Li, Jianqing Li, and Chengyu Liu. Influence of ectopic beats on heart rate variability analysis. *Entropy*, 23:648, 05 2021.

Appendix A

Participant recording information

Participant 1

Remarks

- Aside from the logged moods and activity, could not really derive much meaning from the metrics.

Errors

- Opens the help screen briefly, but does not read the meaning of the metrics.

Discovered bugs

- /

Participant 2

Remarks

- First selects Sunday instead of Monday because of the American date picker, but corrects this
- During instance 2: suggests that the person should take a longer pause between squats
- During instance 3: selects months instead of weeks in trends. Still has the correct conclusion, but has less resolution over the data.

Errors

- Asks what the metrics mean, but does not notice the button to consult the help screen.
- During instance 2: the participant's conclusion was that the person ate too soon after the workout. However, the correct conclusion is that the person ate too soon **before** the workout.

Discovered bugs

- /

Participant 3

Remarks

- The lag from the timeline is a bit bothering for the participant.
- Thinks people won't understand what the metrics mean without any background knowledge.
- During instance 2: notices a correlation between HRV being long under baseline and exhaustion.

- During instance 3: an increase of weeks and months doesn't lead to an increase in x-labels of the graphs when there is no data. This seemed a bit confusing.

Errors

- During instance 2: tries to get a tooltip when hovering over the metric labels, but eventually finds the help screen.
- During instance 3: does not say anything about the increase in physical activity. The conclusion over the logged moods is correct, however.

Discovered bugs

- Clicks on a lingering activity button which displays NaN values. This is removed in the next session day.
- Scrolling sensitivity too high on the mouse.

Participant 4

Remarks

- First selects Sunday instead of Monday because of the American date picker, but corrects this

Errors

- During instance 1: only concludes stress is due to the school project itself, doesn't delve further into the logs or metrics.
- During instance 2: only says that the person went too far with squatting due to seeing a high heart rate and an exhaustion mood, doesn't really look deeper into the stress response and the metrics, seems confused by them.
- During instance 3: doesn't concretely say whether the person succeeded in improving his health or not. Also selects four months instead of seven week in trends. This results in more or less the same, though with less resolution.
- Does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 5

Remarks

- Needed a lot of additional clarification about what the participant needed to do
- During instance 1 and 2: the participant could not see how the data was supposed to help him in finding the cause of stress

Errors

- Selects Sunday instead of Monday because of the American date picker. Also selected the day of the current week instead of the previous one.
- During instance 1: could only conclude that the cause of stress was working on school. Participant didn't click on the moods nor the stress response.
- Does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 6

Remarks

- /

Errors

- Selects Sunday instead of Monday because of the American date picker. Also selected the day of the current week instead of the previous one.
- During instance 1: could only conclude that the cause of stress was working on school. Participant didn't click on the moods nor the stress response.
- Does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 7

Remarks

- /

Errors

- Does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 8

Remarks

- During instance 2: looks back at trends in order to find extra correlations. This is not really an error, however.

Errors

- During instance 1: navigates to trends first and can't derive anything from here. However, the participant goes to the timeline afterwards and finds the cause of stress.
- During instance 2: tries to get a tooltip when hovering over the metric labels, but eventually finds the help screen.

Discovered bugs

- /

Participant 9

Remarks

- During instance 1: couldn't derive much useful information from the timeline
- In the activities: averages were rounded, which led to uncertainty. Therefore, there was insufficient data to from a correct conclusion.
- During instance 3: an increase of weeks and months doesn't lead to an increase in x-labels of the graphs when there is no data. This seemed a bit confusing. Also expected more data, and switched to months to see if there was more data. The participant concludes that there isn't enough data to form a thorough analysis.

- Seems to understand the metrics

Errors

- During instance 1: navigates to trends first and can't derive anything from here. However, the participant goes to the timeline afterwards and finds the cause of stress.
- During instance 2: tries to get a tooltip when hovering over the metric labels, but eventually finds the help screen.
- During instance 3: despite not much data being available, the participant doesn't mention the positive trends from the available data. The participant is probably confused by the unavailability of data from previous months, while only the last seven weeks mattered according to the expected flow.

Discovered bugs

- /

Participant 10

Remarks

- Uncertain about the purpose of the stress responses
- Uncertain about what 'healthy' metrics are in comparison to the general population.
- During instance 1: explicitly says that moods are sufficient to determine the cause. The participant finds it a bit trivial.
- During instance 3: selects the correct amount of weeks, but also wants to look over a longer period to compare preceding months. However, no data is available, but the participant can form the correct conclusion from the available data
- Prefers the calendar item to be clickable.

Errors

- Wants to know the meaning behind HRV, but does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 11

Remarks

- Uncertain about the purpose of the stress responses
- During instance 1: explicitly says that moods are sufficient to determine the cause. The participant finds it a bit trivial.
- During instance 2: aside from noticing that there isn't anything particular to heart rate, the participant also notices that a high heart rate is reached in a relatively short time. The participant also says that insufficient pauses were taken between squatting.
- During instance 2: notices the help screen after not noticing it in instance 1, but thinks it is too complicated to understand for average users. The participant says that there is a difference between reading and understanding. Also there is a lot of metric data which could be confusing.

Errors

- Wants to know the meaning behind HRV, but does not notice the button to consult the help screen.

Discovered bugs

- Notices that tooltips with wrong values are shown when hovering over the data lines in the metric rows. However, the participant fixes this by zooming in. Still, it shouldn't display wrong values when zoomed out.

Participant 12

Remarks

- During instance 1: thinks a baseline is missing in the cEDA row, cannot tell what is high or low.
- During instance 1: in addition to the expected conclusion, the participant thinks that the person began too late on the school project. Judging from the sad mood, the participant thinks the person is afraid he won't finish it on time.

Errors

- Forgets that high HRV is good and thinks the opposite, however corrects himself.
- During instance 2: thinks HRV below baseline is due to being unable to handle stress well. However, HRV below baseline is just inherent to intense physical activity.

Discovered bugs

- The user study with this participant was conducted on another week, but the data was not moved. Because of this, the participant had to go two weeks back on the calender instead of one.

Participant 13

Remarks

- Needed clarification of how timestamps needed to be chosen in the timeline
- During instance 1: does not understand why precisely a stress response was detected because the metrics don't change much at that moment.
- During instance 1: thinks that a decrease in heart rate is paired with a good mood rather than a sad one.

Errors

- First selects Sunday instead of Monday because of the American date picker, but corrects this
- During instance 1: for some reason thought a sad emoji was a happy emoji (looked too fast over it?)
- During instance 1 and 2: misses the causes of tress. Doesn't press on any of the timeline buttons.
- Wants to know the meaning behind HRV and cEDA, but does not notice the button to consult the help screen.

Discovered bugs

- /

Participant 14

Remarks

- During instance 3: notices that moods aren't logged consistently. However, the participant still makes the correct conclusion from the data.
- Prefers the calendar item to be clickable.

Errors

- Wants to know the meaning behind HRV and cEDA, but does not notice the button to consult the help screen.

- During instance 1: does not click on the first logged mood, so the participant cannot find the cause of stress.
- During instance 2: could not click on the moods because the resolution was too low; the participant didn't zoom in.

Discovered bugs

- /

Participant 15

Remarks

- Tries out everything first, which is positive. This way the participant doesn't miss the help screen button like many before.
- During instance 3: thinks there is no good order in the stacking of mood types.

Errors

- Despite having opened the help screen, the participant still wondered what some of the metrics stood for (went too fast over it?)
- During instance 1: doesn't click on moods and therefore doesn't find the direct cause of stress. The participant thought HRV was a cause of stress, but it's an indicator.
- During instance 2: misses the second logged mood and therefore can't find the cause of stress. The participant also goes a bit too fast over the timeline.

Discovered bugs

- HRV comparison in activity: beats per minute unit is written instead of ms next to a value. The participant noticed this. The unit was updated later on.

Participant 16

Remarks

- /

Errors

- Wants to know the meaning the metrics, but does not notice the button to consult the help screen.
- During instance 3: selected months instead of weeks in trends. The participant does not notice a correlation between the line graphs.

Discovered bugs

- Clicks on a lingering activity button which displays NaN values. This is removed in the next session day.
- Because points are needed to initialize baselines, a tooltip with a value was shown when hovering near those points on the timeline. This wasn't supposed to happen.
- Dragging with the timeline showed no data on other days. This was fixed during the next day of conducting the study.

Participant 17

Remarks

- /

Errors

- Wants to know the meaning behind HRV and cEDA, but does not notice the button to consult the help screen.
- Started first with instance 2. Says that squatting and the high heart rate while squatting are the causes of the stress. This is incorrect; the squatting inherently isn't stressful, but it's digestion while squatting.
- During instance 1: thinks the cause of the stress is the school project, but doesn't click on the mood buttons for clarification.
- During instance 2: wants to look up another squatting session in the timeline for better comparison. However, the data from this other session is injected in the database and not meant to interact with.
- During instance 3: selected months instead of weeks in trends. Only looks at the logged moods and says the rest of the metrics aren't important, which is not the case.

Discovered bugs

- Dragging with the timeline showed no data on other days. This was fixed during the next day of conducting the study.
- During instance 3: postgres connection crashes for unknown reasons.

Participant 18

Remarks

- During instance 3: does not have any remarks on the heart rate and HRV graphs, thinks they are not that decisive.

Errors

- Doesn't really inspect the metrics that much, doesn't consult the help screen for more info.

Discovered bugs

- /

Participant 19

Remarks

- Tries out everything first, which is positive. This way the participant doesn't miss the help screen button like many before.
- During instance 3: expected more data for a better comparison

Errors

- During instance 1: thought HRV and heart rate were the same thing
- During instance 1: first thinks the answer lies in trends, which is not the case.
- During instance 1: concludes that stress comes due to a spike in heart rate but this is an effect and not a cause. The participant also doesn't click on any of the mood buttons.

Discovered bugs

- During instance 3: postgres connection crashes with a high amount of weeks/months. Possibly due to zero division when calculating averages of no data.
