INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Anxolotl - An Anxiety Companion App

Nuno Miguel Simões Gomes

Licenciado em Engenharia Informática e de Computadores

Dissertação para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Matilde Pós-de-Mina Pato PhD
André Ribeiro Lourenço PhD

Júri:

Presidente: Carlos Jorge de Sousa Gonçalves PhD

Vogais: Rui Manuel Feliciano de Jesus PhD
Matilde Pós-de-Mina Pato PhD

October, 2022

INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Anxolotl - An Anxiety Companion App

Nuno Miguel Simões Gomes

Licenciado em Engenharia Informática e de Computadores

Dissertação para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Matilde Pós-de-Mina Pato PhD
André Ribeiro Lourenço PhD

Júri:

Presidente: Carlos Jorge de Sousa Gonçalves PhD

Vogais: Rui Manuel Feliciano de Jesus PhD
Matilde Pós-de-Mina Pato PhD

October, 2022

Concluding a major accomplishment is always a mixed bag, while I was presented a formidable challenge that pushed me to my limits, I am in equal parts sad for the conclusion of a great journey and ecstatic for what accomplishments lie ahead of me. But this journey was not made alone, this strenuous journey could not be finished alone. I would firstly like to thank everyone that helped through this.
Lastly, I would specially like to dedicate this work to my grandmother Beatriz Simões, who always had an unbreakable trust in me, and inspired me with so much determination every single time we were together.

Acknowledgments

An effort of this magnitude can not be achieved alone, and all through the way, I relied on help from multiple people. While in the end, this is my work, I can not help but feel grateful to those who helped me reach this poinnt.

I would like to start by thanking for the opportunities and resources that were provided to me by Instituto Superior de Engenharia de Lisboa - Instituto Politécnico de Lisboa. I also would like to thank my supervisors Matilde Pato and André Lourenço for their patience, support, tireless effort in helping me produce my best work, and the consistence in always guiding me to achieve more and better than I expected.

University-wise, I would once again like to thank ISEL and Matilde Pato for giving me the chance to participate in the PlanoA-Saúde research project. While I played a small part in it, it was the nudge I needed to bump into the biotechnology area, which led to this thesis. I would also like to thank David Albuquerque and Miguel Luís, two colleagues that made this journey with me, from start to finish, and always had my back, even when I was not able to perform at my best level.

Lastly, I would like to thank my family, Maria Teresa Gomes, Vitor Miguel Gomes and my sister Sofia Gomes for always having my back, and having a tireless trust in me and my work, even on times that I did not believe in my own competence. I also would like to thank my close friends João Martins, Bruno Santos, Frederico Oliveira and Pedro Duarte for being by my side throughout this stressing, time-consuming and exhaustive period of my life, while always providing a safe harbour in a stormy night.

It certainly takes a whole village to help someone get through the rigours of an academic degree, but that is one of the reasons that makes the conclusion so sweet.

Abstract

World Health Organization referred that common mental health disorders were the biggest contributors to global disability during the year of 2015, with anxiety disorders occupying the 6th position. Currently, anxiety disorders have high prevalence in society, and present early symptoms that are suited to be detected. With this thesis, we intend to produce a system capable of detecting the anxiety disorder early symptoms before the onset of the full range of symptoms. Additionally, we want to give another option to people already affected, in the form of monitoring their mental health, and the ability for them to react to their anxiety state quickly. Herein, we are introducing a mobile health system — Anxolotl, that can detect and classify multi class anxiety levels and detect binary panic states. Our solution is composed by: a datacenter, intended to store anonymous physiological data and applying the machine learning models; a message broker, aiming to provide scalability and decoupling to the system; and, finally a mobile app, which will work in tandem with a wearable to capture physiological data. The app is able to track and monitor, on a daily basis, its user's anxiety and panic levels, filtering when the data is unreliable based on activity. It also presents the users with guided breathing exercises for multiple mental health scenarios as well as some guided meditations, in an effort to help its users. The Anxiety Engine model provided a 92% accuracy and 90% f1-Score in classifying multi-class anxiety levels, training and testing with a dataset containing 124 entries, and our binary Panic Engine had an accuracy of 94% and a f1-Score of 94%. Both these scenarios were mainly achieved by using heart rate data, activity context was also used in some scenarios. The code for these models is available at `https://github.com/nunogoms/Anxolotl-engines`.

Keywords: mobile health; biosensors; wearables; anxiety and panic classification; machine learning; mental health monitoring system; biomarkers; physiological signals

Resumo

A Organização Mundial de Saúde apresentou as perturbações mentais como os maiores contribuintes para incapacidade global em 2015, com os distúrbios de ansiedade a ocuparem a sexta posição. Distúrbios de ansiedade têm um alta prevalência na sociedade, e apresentam sintomas precoces que podem ser detetados. Nesta tese, produzimos um sistema capaz de detetar sintomas de distúrbios de ansiedade antes que a doença se instale por completo. Adicionalmente, queremos dar outra opção a portadores, monitorizando o seu estado mental e oferecendo a hipótese de tratarem dos seus níveis de ansiedade antes que apareçam mais sintomas. Aqui introduzimos um sistema de saúde móvel, entitulado de Anxolotl, que pode detetar e classificar níveis de ansiedade multiclasse e detetar níveis binários de estados de pânico . A nossa solução é composta por: *datacenter*, com o objectivo de guardar dados fisiológicos anónimos, e aplicar modelos de aprendizagem automática; *broker* de mensagens, que irá providenciar escalaabilidade e habilidade de desacoplamento no sistema; aplicação móvel, que funcionará em conjunto com um *wearable* para capturar dados fisiológicos. A nossa applicação é capaz de detetar e monitorizar diariamente, os níveis de ansiedade e pânico do utilizador, filtrando dados dúbios com base em atividade física. A aplicação também apresenta múltiplos exercícios de respiração guiados, bem como meditações acompanhadas para vários cenários de saúde mental. O nosso modelo de deteção de ansiedade foi capaz de apresentar uma precisão de 92% e um *f1-Score* de 90% na classificação de ansiedade multiclasse, treinando com um *dataset* com 124 entradas, enquanto que o nosso modelo de deteção de pânico apresenta uma precisão de 94% e um *f1-Score* de 94%. Estes valores foram atingindos utilizando maioritariamente dados de ritmo cardíaco. O código dos modelos está disponível em `https://github.com/nunogoms/Anxolotl-engines`.

Palavras-chave: monitorização móveç; biossensores; wearables; classificação de ansiedade e pânico; aprendizagem de máquina; sistema de monitorização de saúde mental; biomarcadores; sinais fisiológicos

Contents

List of Figures

List of Tables

List of Listings

Acronyms

ACC	Accelerometer. 6, 15, 18, 21, 26, 46, 69
AES	Advanced Encryption Standard. 29
AI	Artificial Intelligence. 13, 19
ANS	Autonomic Nervous System. 7
APD	Anxiety Phases Dataset. 46, 47, 48, 49, 50, 52, 55, 56, 58, 60, 64, 66, 70, 71
API	Application Programming Interface. 27, 72
Auth	Authentication. 28
BLE	Bluetooth Low Energy. 9, 24, 26, 27, 30
BP	Blood Pressure. 15
BR	Breath Rate. 19, 21, 26, 50
CBT	Cognitive Behavioral Therapy. 4, 5
CSV	Comma-Separated Values. 44
DTO	Data Transfer Object. 43
ECG	Electrocardiogram. 6, 7, 14, 15, 16, 17, 18, 19, 47, 50, 56, 70
EDA	Electrodermal Activity. 6, 7, 8, 15, 16, 17, 18, 19, 21, 26, 47
EEG	Electroencephalogram. 15, 69
EMA	Ecological Momentary Assessment. 17
EMG	Electromyogram. 6, 16
FSStorage	Flutter Secure Storage. 29, 30, 31

GAD Generalized Anxiety Disorder. 2, 4, 5, 6, 7, 8, 18, 19, 21, 35, 39, 44, 76
GSR Galvanic Skin Response. 16, 17, 20

HF High Frequency. 8, 36, 50, 51, 58, 70
HR Heart Rate. 4, 5, 8, 15, 16, 17, 18, 19, 20, 21, 27, 36, 46, 51, 56, 57, 69, 72, 77
HRV Heart Rate Variability. 6, 8, 9, 15, 17, 18, 19, 21, 36, 43, 46, 50, 51, 52, 58, 60
HTTPS Hypertext Transfer Protocol Secure. 38, 73

IBI Interbeat Interval. 8, 27, 33, 42, 48, 49, 50, 51, 56, 57, 70
IoT Internet of Things. 23, 39

KNN K-Nearest Neighbors. 20

LF Low Frequency. 8, 36, 50, 51, 58, 70

ML Machine Learning. 9, 13, 19, 20, 24, 25, 42, 44, 54, 57, 58, 60, 63, 66, 71, 74, 76, 77
MQTT Message Queuing Telemetry Transport. 39

PA Panic Attack. 5, 7, 9, 18, 19, 20, 35, 67, 70, 71
PD Panic Disorder. 2, 4, 5, 6, 7, 8, 18, 19, 20, 21, 39, 44, 76, 77
PNS Parasympathetic Nervous System. 7, 8, 36, 50
PPG Photoplethysmography. 6, 7, 9, 17, 18, 20, 26, 50, 69
pub sub model Publisher Subscriber Model. 39

RF Random Forest. 20
RPM Remote Patient Monitoring. 13
RR Respiratory Rate. 16, 18
RSA Rivest–Shamir–Adleman. 29

SMOTE Synthetic Minority Oversampling Technique. xv, 53, 54, 59, 60, 65
SNS Sympathetic Nervous System. 7, 8, 18, 50

SO2 Oxygen Blood Saturation. 15
STAI State-Trait Anxiety Inventory questionnaire. 46, 48
SUDS Subjective Units of Distress Scale. 47, 56, 57
SVC Support Vector Classifier. 65, 67
SVM Support Vector Machine. 20

TLS Transport Layer Security. 24, 25, 39, 40, 73, 75

VLF Very Low Frequency. 8

1 Introduction

Common mental disorders were the biggest contributor to global disability during the year of 2015, with depression disorders being the first overall, and anxiety disorders being the sixth [52]. Stress, nervousness and anxiety affect most people at some point in their lives, but the regularity at which that happens is one of the key points in classifying such problem as a disease. Focusing on the anxiety disorders, they are the most common type of mental illness in the world, affecting 264 million people worldwide, as of 2017, with an increase of 14.9% per decade [10].

As of late, the wearables market has been slowly improving and becoming widespread, and their yearly improvements still rely on providing more sensors and techniques to gather data. With this background, we were set on devising a system able to capture physiological data using a wearable, and then extracting value from that data by providing a user facing mobile app that would present the users with their mental health metrics. In an effort to provide more than a simple tracking system, we also intend to provide some kind of wellbeing exercises, so users can manage their mental health in a simple and easy to achieve manner.

Our intended system will heavily focus on detecting and classifying mental health metrics, such as anxiety levels and the existence of panic. On that note, we participated on a Challenge on Detection of Stress and Mental Health using Wearable Sensors promoted by the IEEE Engineering in Medicine and Biology Society. Our team's results were not remarkable from a nominal accuracy value, which was around 64% accuracy at detecting stress with a pre-analysed provided dataset. But on a relative scale, it was the best result among all the talented contenders. Following up on that, we were able to achieve the first place with our stress classification algorithm and article. The developed algorithm is available at https://github.com/matpato/CfP-Workshop-and-Challenge-Wellbeing.

1.1 Mental health context

In the common mental disorders category, there are two main diagnostic subcategories: depressive disorders and anxiety disorders [52]. The spectrum of mental disorders is much larger than this, but as far as research goes, these two seem to be the most prevalent. They can be seen as a continuum, starting from the root cause of fear and stress until the end of the continuum which would be depression [6]. In both cases, the severity will oscillate between mild to severe, and the duration is variable in all the cases, from a few months to a lifetime, called a chronic condition.

Depressive disorders are characterised by sadness, loss of interest or pleasure, feelings of guilt or low self-worth, disturbed sleep or appetite, feelings of tiredness, and poor concentration [52]. The duration of depression varies wildly, and it can be once in a lifetime occurrence or a recurrent issue. It can also be marked by depressive episodes, with varying degrees of severity, and it is not abnormal for those episodes to intercalate with regular mood.

Anxiety disorders refer to a group of mental disorders characterised by feelings of anxiety and fear, including: **Generalized Anxiety Disorder (GAD)**, **Panic Disorder (PD)**, **phobias**, **social anxiety disorder**, **obsessive-compulsive disorder (OCD)**, and **post-traumatic stress disorder (PTSD)** anxiety disorder [52]. The severity and duration of symptoms has high variation, but they tend to manifested as a chronic condition. This factor, coupled with curiosity regarding the subject, made GAD and PD the target diseases in our study. Data from the Netherlands Study of Depression and Anxiety showed that after 2 years, 23.5% of patients in remission experienced a recurrence in their anxiety disorder [41]

As of late, more research is conducted into recognising anxiety disorders affecting the general population, and this is due to many factors, one of them being the aforementioned scope of these disorders. Another is the existence of subtreshold cases in PD, which are cases that do not classify as clinical PD due to a failure in meeting the full clinical criteria. Even though these cases are not recognised, they have been shown to predict the development of full PD as well as other mental disorders, such as Generalized Anxiety Disorder (GAD) or major depression [23]. Anxiety also presents a very high socio-economical cost in the population. Medical workups can be extensive and delay treatment of symptoms. Anxious patients may undergo unnecessary tests or procedures and be referred to various medical specialists before an appropriate psychiatric consultation is sought, or a diagnosis is made [2].

Anxiety is one of the most pervasive and ubiquitous human emotions, in all cultures [45]. It is considered a basic negative emotion, such as sadness, anger, worry or fear. Anxiety, fear and stress all share similarities and might even overlap to some extent, but they are different states: In the case of stress, it defines a bodily reaction, usually while reacting to a challenge or a demand, such as in an encounter with a tiger. Nowadays, it is unusual to find tigers in a person's day-to-day life, and so, it is more prevalent in the case of deadlines or responsibilities, and it relies on helping humans to be better prepared to deal with such events, using biological changes to better face a threat. Stress has a clear cause, which is called a stress-causing factor or a stressor, such as the tiger in aforementioned example. Fear also shares some similarities to stress, but it is classified as an emotion and can trigger a stress response. It also is associated with danger or insecurity with the main characteristic of being focused on immediate present danger.

Anxiety, by contrast, corresponds to a state of uncertainty [56], and it is more closely related to a future-oriented mood state associated with preparation for possible, upcoming negative event [6]. Anxiety on itself is not a problem, it also might help overcome some challenges when present in low amounts, such as the worry when preparing for an exam, or a presentation. It is experienced through both mental and physical symptoms. The state of anxiety is accompanied by feelings of nervousness and tensions, as well as worries and intrusive thoughts. It is also typical to experience signs of bodily activation, sometimes described as "fight-or-flight" response, such as a pounding heart, perspiration, and gastric disturbance [56].

As stated before, anxiety is a normal reaction to danger and fear, it might be uncomfortable, but it was a response designed over millions of years of evolution to help us survive and thrive. Therefore, the existence of anxiety disorders might come as counterintuitive, but in the same way stress can be a disorder, so can anxiety. Disorder-like anxiety happens when it exists without a clear identifiable reason or danger and when it affects a person in their day-to-day life. When it happens without a clear identifiable reason, it is particularly tricky, since by then the anxiety might not be time-limited. An anxiety disorder is when anxiety symptoms exist for an extended period of time, usually equal or longer than six months.

Based on these 6 months, it is easy to recognise how easily anxiety disorders can take over healthy living years away from people. There are multiple diseases that fit onto the umbrella term of anxiety disorders. This research work aims to take an approach towards Panic Disorder and Generalized Anxiety Disorder. Both of these diseases present multiple symptoms, which most commonly are somatic (physiological), psychic, behavioural or cognitive.

Starting with GAD, which is as close to the idea of problematic anxiety as there is, it can be defined as a dysregulation of the anxiety and worrying system. Patients with GAD often present a lifelong history of generalised anxiety [12]. This disorder is also characterised by a difficulty in controlling worrying, on why it happens and when it happens, as well as a struggle in engaging in relaxing activities.

Anxiety disorders tend to run a chronic course, with symptoms fluctuating in severity between periods of relapse and remission in GAD [2]. Some of the most common symptoms are [2, 14]:

- Somatic Symptoms
  - Tremors,
  - Palpitations (increased or irregular Heart Rate (HR)),
  - Dizziness and Nausea,
  - Shortness of breath,
  - Sweating, and
  - Muscle tension.
- Psychic Symptoms
  - Difficulty concentrating,
  - Nervousness,
  - Insomnia, and
  - Constant worry.

Regarding therapeutics, without delving into medicine, there is some psychological help that is proven to have a very positive effect on patients. Some offerings are guided self-help, which focuses on some material recommended by a medical professional to be applied by the patient alone; Cognitive Behavioral Therapy (CBT), one of the most effective treatments for anxiety disorders, presents itself as a guided method to cope with negative, anxious and intrusive thoughts; and lastly, applied relaxation, which focuses on relaxing muscles when a situation that leads to increased anxiety arises.

Anxiety disorders have a tendency to not happen alone, since the cerebral activity present in all the mental disorders is related and can overlap to some extent. This is particularly true for GAD, which most often co-occurs with other disorders such as PD and social phobia [12], it also presents a strong relation to chronic stress. Panic disorder is a heterogeneous psychiatric disorder characterised by recurrent, unexpected episodes containing at least one Panic Attack (PA), the hallmark of the disorder, also marked by persistent concern about additional PAs or their consequences, and/or behavioural modification as a direct result of the PAs; up to 70% of the patients with PD also have agoraphobia [4].

Currently, agoraphobia is classified as a complication of PD, since its symptoms can come from the association of PAs with places or situations where they occurred. It is defined as fear of places where it might be difficult or embarrassing to escape if a panic attack should occur (crowds, on public transport, or in closed spaces). Fear of being alone is also common [2].

Another reason for the choice of PD is its close relationship to GAD, since panic is the most severe form of anxiety. The main symptom of PD is the panic or fear of attacks which can happen regularly for no apparent reason. Their erratic nature makes their occurrence often frightening and distressing, and support after and during one might be helpful. They can be defined as attacks of sudden onset, with only physical (somatic) manifestations, in which the symptoms include : (1) palpitations (increased or irregular HR), (2) sweating, (3) tremors, (4) dry mouth, (5) dyspnea (Shortness of breath), (6) dizziness and Nausea, (7) hot Flushes or Chills, (8) hyperventilation, and (9) trembling [2, 4].

Focusing on PD therapeutics, medicines are more widely used in this kind of disorder, but there is space for psychological therapies as well. Most of the therapies that are used in the treatment of GAD can also be used in PD, with a special focus on CBT and applied relaxation. While the medical field is focused on making the symptoms weaker, the therapies are aimed at avoiding and coping with the PDs. Despite the existence of several therapeutic options for these symptoms, PD treatment continues to pose a challenge because the crucial question about what is the best intervention for each individual patient remains elusive [4].

The production of smart devices to help individuals monitor components of their health has been on the rise the last few years [20]. Given that the presence of smartphones among the population is almost universal, these two tools could be used as a way of bringing comfort and quality of life to people suffering from anxiety disorders.

Nowadays, multiple sensors can be found in inexpensive wearable devices, that have the distinct advantage of being very comfortable to wear, taking good readings and not being invasive at all. Currently, there are multiple wearables on the market with an affordable price tag that contain sensors such as: (1) Photoplethysmography (PPG), (2) Electrocardiogram (ECG), (3) Accelerometer (ACC), (4) Electromyogram (EMG) or (5) Electrodermal Activity (EDA), usually in a combination of these sensors. The presence of location services in almost all smartphones might also be interesting in cases such as the agoraphobia case. Using Heart Rate Variability (HRV) as an example, it can be extrapolated from data captured from both an ECG or a PPG, annd HRV was identified as the most useful physiological metric for detection of stress and anxiety [20]. With such a vast selection of sensors, and recently found formulas to extrapolate more data from a single measurement, consumer wearables present a very interesting point of entry to monitor these illnesses.

Resuming the presented arguments, such as the existence of untracked subtreshold cases, the case for personal targeting bringing better results, the ever-increasing incidence of anxiety disorders, and the apparent possibility of tracking a lot of GAD and PD symptoms and signals with inexpensive and already existing tools. We believe there is a possibility of creating a system to track, detect and possibly predict baseline anxiety for GAD, as well as panic attacks for PD, and possibly even stressors. Such system could bring a new way to keep track of symptoms, state and evolution of these diseases in patients, while trying to overcome current challenges in the existing solutions such as stigma and low accessibility. This solution would bring control to patient's lives, while not taking control away from the medical professionals, since they would be able to use the recorded data to understand the progress of their patients. The system would be focused on a mobile application and a wearable, with the application being mainly focused on the data presenting, and providing psychological help to the patient and the wearable would focus on sampling data.

1.2 Background of the physiological measures

Nowadays, Computer Science is applied in some way or another in multiple aspects of our daily lives, from the technological devices we use, to our waste management and health check-ups, its impact on modern life can not be understated. On the contrary, the overlap between Computer Science and all the faculties that it is applied is minimal in most cases, and for this reason we present here a small theoretical background for the impact of physiological measures in the illnesses we aim to target.

Physiological data relations Wearables nowadays contain multiple sensors, which we will present in the next chapter, but some of the most commonly used are the PPG, ECG and EDA. To target both GAD and PD, we need to understand how this illnesses impact the results captured by those wearables, and some of their features. The effects on this physiological data will be investigated under the motto of anxiety detection and panic detection.

The human nervous system is divided into two main sections: the Central Nervous System, made up of the brain and spinal cord, and the Peripheral Nervous System, which incorporates all the nerves that branch off the spinal cord to all the parts of our bodies such as limbs and organs, almost behaving as a relay to our brain and spinal cord. The latter is further divided into the Somatic Nervous System, responsible for collecting sensory nervous data and touch sensory data, and the Autonomic Nervous System (ANS), which is charge of controlling involuntary responses to *stimuli*, such as salivating when seeing food or the pupil's dilation and contraction in reaction to light.

The ANS is the part of our nervous system that is most closely linked with anxiety and panic responses, given their involuntary nature. It's two main divisions are the Sympathetic Nervous System (SNS) and the Parasympathetic Nervous System (PNS), while also having a third division, which we are opting to leave out in an effort of simplification. Firstly, the SNS is activated during a "fight or flight" situation, in which a person is presented with physical danger or mental stress, and it can turn off non-essential functions such as digestion. The PNS is used during the remainder of the time, also known as a resting state, and unlike the SNS, allows humans to have some control over certain traits, such as facial expressions and breathing. These two systems coexist in a balance, and the increase in activity from one of them necessarily implies a decrease in activity of the other.

Now, both anxiety and panic states imply heavier SNS control, since these states are responses in a "fight or flight" situation. Research suggests that in the case of patients of PD and anxiety disorders, their SNS activity is no different from ordinary people during resting periods. [53]. It is also important to reiterate that Panic Attacks are considered the most severe form of anxiety, so while they are not exactly the same, given the fulminance of PAs when comparing to anxiety symptoms, they are quite related.

Regarding the most used wearable sensors, and their ability to measure SNS activity, which could be correlated with anxiety and PD activity, we firstly have the EDA. It mainly targets the production of sweat by the human body, which is by itself a non-voluntary action, pursued by the ANS. It is one of the few non-invasive measures of sympathetic arousal. If the sympathetic branch of the autonomic nervous system is highly aroused, then sweat gland activity increases, which in turn increases EDA values. EDA has been reported to be more closely related to emotion than HR, and thus is utilised in psychological research, particularly regarding scenarios and assessments of emotional arousal, stress and appraisal [42].

Most wearables, nowadays, do not contain an EDA sensor, and another important vital signal that is controlled by both the SNS and PNS during their different control states is the HR. While not major distinguishable differences happen between each case, upon closer inspections of the HR features, we can see that there is a difference. These features all flourish from the basis of Interbeat Interval (IBI) capturing, which refers to the time between each heartbeat. Some of the most used features come from the spectral analysis of the Heart Rate Variability (HRV) and are its Very Low Frequency (VLF), Low Frequency (LF) and High Frequency (HF). Another very important feature of the base HR, is the Interbeat Interval (IBI), which is used to extrapolate the Heart Rate Variability (HRV) from a subject.

Individuals suffering from an anxiety disorder, such as PD or GAD are characterised by chronically low Heart Rate Variability (HRV) compared to healthy individuals during resting state conditions [19]. Furthermore, there are physiological ways to distinguish between them, such as a lower than normal LF/HF ratio in the case of PD [29], and the opposite in the case of GAD, specifically a higher than normal LF/HF ratio [54].

According to [24], in the case of PD, their baseline VLF band tends to be lower compared to controls, as well as sligtly bigger band in the case of HF. In periods of mental tasks, the LF/HF ratio presented an higher value than the controls, as presented before.

1.3 Objectives

The presented thesis focuses on trying to implement a simple solution to a very nuanced problem, which is the management of mental health and follow-up of mental illnesses, namely GAD and PD. This solution focuses on trying to take advantage of consumer grade wearables, which are already very present worldwide, and using them to give more control to patients in how they manage their mental health. The big focus here is to provide another support tool, one mainly used by GAD prone people, to keep track of their data, and allow them to intervene before an acute crisis settles in.

This solution starts on the usage of wearables, initially the HBand SP-2D-M, which will be used to capture physiologic data in a non-intrusive manner. The data capture is mainly focused on heart activity features, such as the Heart Rate (HR) and the Heart

Rate Variability (HRV), which will be captured using a Photoplethysmography (PPG) sensor. Since that data can not be distinguished from day-to-day stress, we will use physical activity data captured from the wearable, labelled as quantitative data (rest, sleep, exercising, etc) to give context to the measurements to try and avoid false positives. This data will be captured during cyclic intervals, to guarantee a good amount of baseline data.

The wearables will connect to the app *via* Bluetooth Low Energy (BLE), and use either proprietary services or ideally generic documented services to communicate. Regarding the app, probably questionnaires will also be needed to add more context to the captured data, but these will be used sparingly and in last resort, since they hamper the usage of the application.

With the captured data, we intend to preprocess it on the smartphone to reduce noise, and upon having the full processed data, it will be compared to medical values of what is acceptable and not, and after that it will be fed to a trained Machine Learning (ML) algorithm. This algorithm will be trained with a derivative of HRV, which is already used in some research to predict PAs.

The ML algorithm is intended to be stored in a data centre, which will be away from the application, and will learn from all the instances of the application using anonymous data. After each iteration of learning and testing, we intend to keep updating the application with the best suited models.

In the end, we wish to have a system that can reliably detect anxiety levels and detect and predict Panic Attack as long as the wearable is being used. Ideally, upon the detection of abnormal levels, some exercises will be suggested by the application via notification, and in the detection of a PA, the user will be able to choose which mechanisms to use, such as automatically calling a selected person, buzzing or suggesting breathing exercises. The application is intended to run on the background and auto-start to be as frictionless as possible to use. Some ideas for optional development include giving a key to the app, to allow a patient to share the data stored on the smartphone with a health professional, helping the professional to make a diagnosis. It is important to refer that this solution does not aim to be a medical tool, or a substitute to medical aid, but aims to be a tool for anyone to use with the intent of better handling their mental health.

1.4 Publications

Before this project came to fruition, there was a foray into the world of wearable devices being used to target and track medical conditions called PlanoA-Saúde, published at https://www.planoa.pt/. With the work derived from that project, we presented and wrote an article at the conference 12º simpósio de Informática — INForum 2021, named "A platform for medical remote monitoring of patients with chronic diseases" [16].

As an extension of the work presented here, we also participated on the EMBC 2022 Workshop and Challenge on Detection of Stress and Mental Health Status Using Wearable Sensor, situated at https://compwell.rice.edu/workshops/embc2022. To that end, we wrote and published an article in the 44th International Engineering in Medicine and Biology Conference, by the name of "Anxolotl, an Anxiety Companion App - Stress Detection". This publication is not yet publicly available, as it is set to be published in a book, which we are still waiting for. Meanwhile, the article is available at https://drive.google.com/file/d/1NnmRxG5l16xcVZyT9L3eU0AHvC4yXdxn/view.

We have made a submission to a Special Issue of the peer-reviewed journal Sensors, of MDPI, namely "Recent Developments in Sensors for Wearable Device Applications", with the intent of publishing this thesis' State of the Art to the scientific peer-reviewed journal Sensors. The name of the paper is "A survey on wearable sensors for mental health monitoring", and the authors are Nuno Gomes, Matilde Pato, André Lourenço and Nuno Datia.

1.5 Document Structure

This document is divided into 6 chapters. The first two chapters focus on theory, with the first chapter consisting of the introduction to the thesis, 1, which pertains to the theoretical reasoning for this project, a theoretical background and the expected goals we intend to achieve. The chapter 2 consists in the state of the art and related work.

In the chapter 3, we present a small overview of our system, and then proceed to explain in detail how all of our components are implemented. Since we have a very strong machine learning component, we detail all of our process regarding the developed algorithms in chapter 4.

The last third of this thesis is marked by the results, in which we will present our machine learning results as well as the results that were viable to be quantified using our system, we then proceed to evaluate and discuss the results achieved, and all of this is presented in chapter 5. Finally, to wrap up our work we conclude with future directions to take, as well as the main takeaways from the project in chapter 6.

2

State of the Art

Recently there as been an increase in the usage of technology by the healthcare sector, a more nuanced one than simply in allowing more types of examinations to be run. This arrival came in multiple fronts, be it the usage of wearables, assistance of Artificial Intelligence (AI) and Machine Learning (ML) in decision-making [28], or simply the usage of electronic medical records. Monitoring patients has been an essential part of looking after patients, since performing multiple examinations on patients is the *modus operandi* to find out a patient's illnesses. In theory, an abundance of readily available non-invasive data about a patient should allow medical professionals to better address health issues and possibly improve the targeting of conditions with a smaller usage of hospital resources.

Remote Patient Monitoring (RPM) stands as a way to collect data from an individual, and can be made with invasive sensors, wearables or questionnaires [50]. Regarding AI and ML, many research projects show that combining AI with human judgement produces the best outcomes, but this research is still in its infancy [3] and its viability must be taken with a grain of salt.

During 2017, Vegesna et al. [50] performed a literature review of 62 studies focusing on RPM via non-invasive digital technologies. The technologies mentioned are multiple, yet we will focus on analysing the ones that fit the scope of this project, in this case measurements with smartphones and wearables. The results were evaluated as having positive, negative, or neutral health outcomes, varying according the impact of the intervention in the outcome of the study [50]. According to the evaluation made, wearables had a 50% positive outcome with the rest being neutral, smartphones improved the outcome of the study in 58% of the cases, while having no negative impact on the remainder of cases(neutral). Study designs included both randomised controlled trials, observational studies, and systematic reviews published from January 1, 2005 to September 15, 2015 [50], and while this is a representative study, the technology has improved a considerable amount since then, according to Moore's Law expectations.

As exposed in the presented research, mHealth devices have the potential to decrease the cost of both clinical research and health care just as technology advancements have done in virtually all industries except health care [49]. Within the past decade, the creation of commercially available smart devices and wearable technologies to monitor health has grown exponentially [20], and with it, the use cases for wearables has increased. This study wants to focus on the applicability of wearables in the setting of monitoring anxiety disorders and panic attacks primarily, while possibly taking the care to address any secondary measurements that could relate to this such as the association between anxiety and stress [33].

2.1 The case for wearables

Recovering patient's data while they are going on about their lives is not a novelty in the medical community, some devices are already widely used while keeping that premise, such as the 24 hour Holter monitor for ECG measurement. The future seems to be heading in the way of applying more digital technologies in the medical field, as they are continually being adopted as an additional method for healthcare systems to increase patient contact and augment the practice of preventive medicine [50]. The common place of wearable devices already makes them a great candidate, given that in 2018, one in five individuals in the United States owned a wearable device, most of which were activity trackers [55].

In combination with value-based healthcare systems through telehealth, wearable devices can enable monitoring risk patients intervening diseases at an earlier stage, and reducing healthcare expenditures by means of prediction and prevention of disease [55].

Citing Steinhubl et al. [49], wearable biometric sensors allow for unobtrusive, passive and continuous monitoring, and one of their key characteristics is the ability to seamlessly track and transfer all biometric data into an actionable and informative user interface that can be shared with health care providers and researchers. In the end, the amount of data on record from a single patient, allows healthcare professionals to address their patients according to an individualised (that is, precision or personalised) medicine [49].

On a continuous and personal measurement, the measurements must be made outside the clinical environment using semi-validated, wearable devices that are tolerated by people [20]. Our proposal takes the comfort of use, and non-invasiveness of the sensors in utmost regard, since the constant reminder of being analysed might skew a user's behaviour and data, as well as the discomfort of usage might hinder the long term adherence of the project. According to [34], the optimal sensor location, should ensure accuracy of sensors and minimise visibility. The authors concluded that patients have higher preference and compliance for devices placed on the wrist, as such devices can often be perceived to be clothing accessories. This project relies on having good adherence by the users, since without continuous data from big swaths of time, it is not possible to infer the baseline and consequently detect erratic data values.

The range of metrics that wearables are able to take nowadays is enormous. Multiple sensors from multiple wearables have been tested in multiple settings, and research projects. Some variety of sensors present in the current wearable market are represented here, such as (1) Electrocardiogram (ECG) [20, 34, 55], (2) Heart Rate (HR) [20, 34, 55], (3) Caloric Expenditure [20, 55], (4) Skin Temperature [20, 55], (5) Oxygen Consumption [55], (6) Sleep [34], (7) Respiratory Rate [20, 34], (8) Accelerometer (ACC) [20, 34], (9) Oxygen Blood Saturation (SO2) [34], (10) Blood Pressure (BP) [34], (11) Electroencephalogram (EEG) [20], (12) Electrodermal Activity (EDA) [20], and (13) Heart Rate Variability (HRV) [20] to name a few. Most of these sensors are relatively common and easy to find in a pre-built wearable system, sometimes even bundled in groups of 2 or 3, which makes these devices especially interesting from a biological data capturing perspective.

In all the presented articles [20, 34, 55], the usage of wearables had no negative impact in the outcome of the study. In most of them, the usage of wearables was the tool used for continuous monitoring of patients, and so the benefits far outweighed the drawbacks of said technology. Regarding the presented cases, the discussion is about having a good amount of data to analyse, or having a small dataset, and in this case wearables are a good solution to get big datasets of somewhat good quality.

Some wearables are already in the race to be used in great quality data capturing, for instance an FDA-cleared, wearable multi-parameter vital sign monitoring device records and transmits ECG, HR, respiratory inductance plethysmography, calorific expenditure, posture/activity, skin/core temperature, and oxygen consumption [55]. Regarding results on wearable sensors, experimental studies to monitor physiological signals and ECG for detecting arrhythmia's have been successfully carried out using the presented type of technologies [55].

Giving an example of some off-the-shelves wearables that have quality to capture good quality patient's data, we have the EQ02 LifeMonitor[1] (Equivital Inc., USA) belt which has been demonstrated to accurately monitor Respiratory Rate (RR) and HR, temperature, biometrics, activity, sleep, fitness and psychophysiology through various animal and human studies [34]. Fitbit wearables[2] are also commonly used in research, and with good results, and consistent readings, with the downfall of being more expensive than most consumer products. Ideally, cheap and accessible wearables, such as Xiaomi Mi Band line, would be the perfect product to develop the idea, given the ubiquity amongst wearable users, but their consistency on readings needs more research.

2.2 Value in monitoring stress and anxiety

Lately, a push towards a better mental health maintenance and recognition has been happening. This section will focus on the work that, as of late, provided good results with wearable compatible sensors in measuring stress, anxiety or panic attacks. Given the focus of this work, a good collection of sensors information to measure each of this metrics will bring immense value, since it will allow for more combinations of sensors to be picked. The pioneers in this field are Healey and Picard who showed in 2005 that stress can be detected using physiological sensors [15].

In the mentioned study, from Healey and Picard [18] the objective was to distinguish between three base levels (low, medium and high) of stress in drivers, presenting an accuracy rate better than 97%. The stress addressed here was the stress with a negative bias, namely distress. Four types of physiological sensors were used during the experiment: (1) ECG, (2) EMG, (3) skin conductivity (also known as EDA or Galvanic Skin Response (GSR)), and (4) respiration (through chest cavity expansion). Their algorithm included the mean and variance of the EMG taken in the hand, respiration and the mean of the tachometer HR over one second intervals throughout the drive. In this study, the best correlating signals with stress levels were the mean of the skin

[1] https://equivital.com/products/eq02-lifemonitor
[2] https://www.fitbit.com/

conductivity (.47), followed by the L100 (frequency domain HRV)(.41) and finally the HR (.30). According to this study, using the HR and skin conductivity with intervals of 5 minutes, allowed them to obtain the stress levels accurately 97.4% of the time.

Another study, from 2020, from Han et al. [17] focused on measuring the stress levels from a population of 17 subjects every day and on a laboratory setting, while binarily accessing their stress condition (stressed or not-stressed). This study focused on the usage of three sensors, Photoplethysmography (PPG), ECG and GSR also known as EDA. In a lab setting the study provided a 94.6% accuracy in distinguishing the stress levels on a laboratorial setting, while that figure dropped to 81.82% on an everyday setting. One of the outcomes of this study was finding that in the everyday setting, GSR + ECG group shown the best everyday accuracy, with 90.91% accuracy [17]. Another finding is that the sensors from the wearables tend to perform worse on an everyday setting, since noise becomes a real problem when it comes to measuring data in an ordinary setting.

Finally, an overlook of the current situation in measuring stress, we can see value in all the presented sensors. While HRV and PPG are relatively recent, they are also promising, as HRV was identified as the most useful physiological metric for detection of stress and anxiety [20], it was also observed that HR and GSR were the most regularly used sensory signals because they gave the most promising results and high-accuracy for detecting stress and its levels [13].

Regarding anxiety, Ecological Momentary Assessment (EMA) which stands for continuous measurement, has the potential to shift the anxiety and panic medical branch perspective towards a "precision psychiatry" approach in which intensive time series data is used to tailor the treatment of anxiety disorders to individual patients [41]. The increase in incidence of anxiety disorders worldwide has motivated researchers to develop new and improved technologies to promote well-being and to reduce related morbidity, mortality, and health care costs [10]. Currently, the continuous monitoring of this kind of patients is seen with good eyes, as it is expected to tell about their specific constellation of symptoms, and also the dynamics of those symptoms and the relationships among them [41].

Measuring anxiety and stress overlaps in a lot of ways, and this hinders the distinction between which sensors to use for each subject, or if a sensor can even be tasked with such objective. The overlap happens between depression, anxiety and stress and is in reason motivated by the fact they share a significant number of risk factors and symptoms. Nevertheless, the reason for the association between these psychological syndromes is yet to be established [40]. For all the monitoring, context is likely to be important, since it allows for a better evaluation of the data, and questionnaires can fill the gap as presented in [3]. If questionnaires are to be used, they will need to be use sparingly, as they can cause questionnaire fatigue, undermining the adherence to the project.

A study from 2021, from Chung et al. [5] on reducing symptoms of anxiety with a HRV biofeedback wearable presented good results. Over the course of eight weeks, the patients performed the questionnaire GAD-2, which stands for Generalized Anxiety Disorder Scale with 2 questions, and a maximum total score of 6. Throughout the study, the patients score dropped three points, which is very promising for a no medical-staff approach. The sensor of choice in this study was the ECG, in spite of the PPG due to the higher reliability and accuracy. The portion of the feedback was delivered via the wearable on the chest, on the form of vibrations to regulate breathing.

According to the 2021 literature review from Hickey et al. [20], cardiac activity has been the only objectively measurable physiological parameter of anxiety in the literature; however, respiratory patterns have been reported to robustly indicate cognitive emotional stress. This review also highlights the usefulness regarding the usage of ECG in monitoring anxiety. The presented results call into question the ongoing challenge of validating EDA and ACC, the first one because it does not seem to perform accurate measurements on wearable devices due to the noise on the readings. Regarding the measurement of anxiety, there seems to be special importance in alterations in HR, RR and EDA since they mirror the function of the Sympathetic Nervous System (SNS) [20], which is closely tied to anxiety activity, and so the measurement of these signals is of heightened value.

The monitoring of panic attacks is not a new topic, but the usage of consumer wearables in this field is, having started around the 2010s. Due to the presented arguments, and a certain distrust in the quality of these wearables as data sources, few studies have delved into accessing the feasibility of using them to monitor and possibly apply biofeedback to anxiety disorder patients while using wearables. We present the Table 2.1 with a review of some studies regarding this topic, with two being particularly deeply analysed.

Regarding the detection of PD, Elgendi and Menon [10] presented mixed results from an ECG and its features. This article regards a literature review, and from the reviewed papers that assessed Panic Attacks, two presented a good correlation between PAs and ECG features, while other two did not. An important factor was that none of the studies revealed the electrodes or wearable placement, and that might have impacted the outcomes. From the favourable result studies, the HRV feature of high frequencies seemed to present the better outcome.

Another good job on this topic was a study from Cruz et al. [7]. This study's main objective was the delivery of help in the form of breathing exercises upon the detection of a PA of a subject. The detection of said attacks are done via a wearable (Zephyr BioPatch), and it can sample HR, temperature, BR and HRV via the usage of ECG electrodes. Upon the data capture, it is sent to a mobile app which processes the data, and if a PA is detected happening or about to happen, an intervention is triggered to the user. The biggest finding of this study was the ability to predict and detect PAs with these metrics as well as letting users add undetected PAs with all of the symptoms it triggered.

As of late, an approximation between ML and medical data has been happening. In the physiological area, related to stress, Aqajari et al. [47], developed a Python Toolkit to preprocess and classify stress responses from EDA. The presented classifiers achieved an accuracy ranging from 80% to 97% identifying stress responses from EDA data from different sources of data.

2.3 Detection and Prediction of Acute Symptoms

Systems of ML and AI are not new. They have seen an explosion around the 1980s, and were hampered by the lack of performance in hardware by then. Nowadays, this field of Computer Science has seen a comeback in popularity, and with it, more investigation.

Physiological data can be assessed from an interval of values, that is how it has been done for most time, and still is in some areas. For instance, when a person takes a blood sample for analysis, the values are considered healthy if they fit within a certain interval, and that has worked good for the most part, yet there are outliers. In the case of health, there is promise in the idea of a personalised approach, one which starts from the common intervals, and then takes values from the patient into account to form a personalised approach.

With continuous data from the patients, via the usage of wearables, it is possible to have a steady source of baseline data. Applying a form of ML in this case, to try and detect anomalous data, and then reacting to that information, brings more value to a GAD and PD support system. Given that the research is still quite recent in the area, it is hard to foray exactly into that, but detecting stress already has some laid groundwork.

In the literature review [13], from 2021, it was investigated if the data captured from multiple different sensors present in wearables, was good enough to apply different techniques of ML and recognise the stress in the presence of a stressor. It was observed that HR and GSR were the most regularly used sensory signals because they gave the most promising results and high-accuracy for detecting stress and its levels [13]. The most used classifiers in this case were Logistic Regression, K-Nearest Neighbors (KNN), Random Forest (RF) and Support Vector Machine (SVM), and there was considerable better accuracy when using data from more than one biosensor. It is also important to state that the data used in the reviewed studies was pre-processed, and it does not seem to have had a negative impact on the outcome. Most of the accuracies in the presented stress classification were higher than 90%, which is promising, and solidifies this approach as viable to detect stress.

On the subject of detection and prediction of PD, Cruz et al. [7] was the best subject to do a comparison. It captures the data from the smartphone camera, and sends said data to a data centre. Then it applies point analysis to build feature vectors, and lastly anomaly detection algorithms are used to distinguish between pre-panic and non-panic intervals. Even though the used dataset was small, and the exact algorithms are not presented, it is said that the system performed as it should. Some insight into the fluctuations pre-panic and post-panic are also given, which is a proof of applicability. One of the shortcomings of this paper, is the usage of the smartphone camera to use PPG, which might not be optimal in PA scenarios.

Regarding the usage of ML in the context of physiological data, the literature is quite recent, and almost non-existent when taking into account the mental health diseases and the monitoring using wearables. With those points said, some research has forayed into different single points which will be applied in this project, but not using them all as a cohesive system. Each factor seems to have ground to stand on itself, and we will use this sparse separated information to try and find out if they can all be combined and present more value than the sum of their parts.

Table 2.1: Summary table of included articles which apply wearable devices to perform anxiety related measurements.

Year	Title	Data	Devices	Questionnaires	Setup	Objective	Results
2015	A wearable and mobile intervention delivery system for individuals with PD [7]	Body temp., BR, HRV and HR	Zephyr Biopatch and Smartphone.	-	Smartphone, wearable and data center.	Mobile and wearable system to detect panic attacks as they are happening, with the option to self report.	Panic attacks can be detected as they are happening, via all the used measurements. The physiological signs from a panic attack can manifest until an hour before the attack.
	The challenger app for social anxiety disorder: New advances in mobile psychological treatment [32]	GPS and ACC	Smartphone	-	Mobile App.	A mobile app designed for the treatment of Social Anxiety Disorder, with multiple principles such as gamification, location awareness, notifications, etc.	There is potential, but no experimental data is provided. A big focus on data security is presented on the conclusion.
2016	Time, Frequency & Complexity Analysis for Recognizing Panic States from Physiologic Time-Series [44]	HRV and HR	From a dataset, multiple devices.	-	-	Create models to distinguish between panic states, as panic attack, pre-panic attack and non panic.	Combining time, frequency and nonlinear domain parameters, models can achieve > 90% accuracy when distinguishing between panic states in physiologic time-series data.
2019	Mobile Biofeedback Therapy for the Treatment of Panic Attacks: A Pilot Feasibility Study [30]	HR, HRV and BR	Smartphone Camera	One, proprietary.	Mobile and wearable system to detect panic attacks.	Providing in-vivo biofeedback therapy for panic attacks, shown on the smartphone screen.	The modality of measurement is feasible with a smartphone camera. There is an apparent placebo effect with the measurement procedure, since it acts to stop the attack.
	A mobile application for panic disorder and agoraphobia: Insights from a multi-methods feasibility study [8]	-	-	PAS, ACQ, BSQ, HAM-A, MI, SIGH-A, CES-D	Mobile App.	Developing a mobile app to intervent in the case of symptoms of PD. The focus is training and help the patients cope with flares.	More research is needed, but the overall, usability, user satisfaction, motivational value and acceptance of the app were perceived as high. Self-monitoring *in vivo* was deemed difficult.
	On the use of wearables and biofeedback interventions for people with anxiety disorders [25]	HRV, HR, EDA and BR	Fitbit and Bioharness4.	-	Varies for the multiple devices.	Test prototypes of gamification apps, based on biofeedback, haptic distraction, memory retrieving, stress assist, to assess its effects on stress.	The devices using biofeedback with gamification were able to reduce stress, with a significant drop after a 60 second usage. The different device effectiveness varies from person to person.
	HR variability monitoring for emotion and disorders of emotion [58]	HRV, HR and BR	-	-	-	An analysis on HRV biofeedback, evaluation of emotional disorders and HRV-based emotion analysis and management with wearables.	The physiological foundations of HRV support it as an important tool for emotion study. Implementability of wearables to assess emotions based on HRV can bring interesting outcomes.
2021	Pilot Study on Reducing Symptoms of Anxiety with a HR Variability Biofeedback Wearable and Remote Stress Management Coach [5]	HRV and HR	Lief Smart Patch.	GAD-2 and PHQ-2.	Device that captures data and provides biofeedback.	The feasibility of using this system to decrease overall levels of anxiety disorders including PD in patients.	A research in self reported anxiety of 3 points (range 0-8) and 1 point in Depression (0-8). This study was taken over 56 days.
	Evaluating a Hybrid Web-Based Training Program for Panic Disorder and Agoraphobia: Randomized Controlled Trial [9]	-	-	HAM-A, BSQ, MI, ADS, SF-12.	Web App in conjunction with mobile app.	Experiment to evaluate the efficacy of a guided hybrid web-based training program based on CBT for adults with symptoms of PD.	The results of this study suggest that a significant number of individuals with symptoms of PD can be helped with this intervention, but needs real CBT.

Sensors Breath Rate (BR), Heart Rate Variability (HRV), Heart Rate (HR), Electrodermal Activity (EDA), and Accelerometer (ACC).

Questionnaires Generalized Anxiety Disorder scale, 2 questions (GAD-2), Patient Health Questionnaire, 2 questions (PHQ-2), Posttraumatic Adjustment Scale (PAS), Agoraphibic Cognitions Questionnaire (ACQ), Body Sensation Questionnaire(BSQ), Hamilton Anxiety Rating Scale (HAM-A), Mobility Inventory (MI), Hamilton Anxiety Rating Scale, reviewed (SIGH-A), Centre for Epidemiological Studies Depression Scale(CES-D), German adaptation of CES-D (ADS), and 12-Item Short-Form Health Survey (SF-12).

Terms Cognitive Behavioral Therapy (CBT).

3

System Implementation

Software oriented for physiological capturing used to have a bad reputation, with a lack maintenance, given the old idea of shipping a product as is. On a positive note, those older products were stable, and usually presented a good degree of availability. The recent pandemic shifted those standards, to more technically nuanced, continually supported solutions, with a big emphasis on IoT devices, as well as the application of new or not already established technologies such as data streaming.

We took the liberty of taking a page from these recent developments in the area, and try to aim to the same level, while taking our limitations into account, the biggest one being the quite small team. We also need to have in mind the proof of concept nature of this project, so we do not end up with a Herculean task in our hands.

Given those arguments, we still want to present a solution that can stand its ground, and to that end, we intend to implement it to be safe, reliable, scalable, and easy to use. Of course all of these attributes do not come often, and we intend to make this project to a standard near those of medical applications, in the way that we keep the data safe at all times, have a scalable and reliable system, and ideally keep the project easily maintainable over its lifetime.

While our system will not be subjected to the amount of testing as one of professional systems, we can keep our hopes alive by relying on practical testing. We will try to maintain a safe connection since the data leaves the wearable device, until it reaches our database. After that, the data we send is calculated by our system, so that is not as much of a concern, yet it will still be protected.

Next, we will present our idealised architecture, and the recipe to follow to reach our intended system. That theoretical grounding is made as a technology-agnostic approach to our problem, that could lay the groundwork for the implementation itself. We will also present the most important features of each part, and how they fulfil certain roles in respecting the aforementioned values.

The last section, will be used to detail our implementation of the system, regarding its technologies, as well as the choices taken to support them. We will iterate over every part of our system in detail such as the mobile app, the message broker and the datacenter.

3.1 Anxolotl System

The architecture of this system is one defining part of achieving our self proposed goals in regard to developing a sustainable system that can be upgraded and maintained. We will start by defining the guidelines of our system, and then we will try to apply them to the best of our abilities.

Our core values in this regard would be security, modularity, some fault tolerance, as in one system crashing, not causing the system to lose data and finally the possibility for scalability aimed at future iterations. Regarding our user experience, we would like it to be intuitive, seamless, and ideally not needing the user's constant assistance, such that it could be a plug and play experience.

To this end, we are splitting the system into three parts, which can be visualised in Figure 3.1:

- Anxolotl App – Includes the mobile app and the wearable,
- Broker – Just regarding the message broker, and the
- Datacenter – The server running the ML algorithms and a database.

Starting with our system's entry point, which is the wearable, it is going to connect to the application. The application main job is to receive and lightly process the data, and having a reliable connection to the wearable. The app should also not lose data in the case the broker is down for any reason. We are using Bluetooth Low Energy (BLE) to communicate with the wearable, since that was the only reasonable method available. Lastly, our app will also receive the data from the message broker and present it to the user. After receiving data from the wearable, our app will need to send the data which is encrypted *via* Transport Layer Security (TLS) to the message broker.

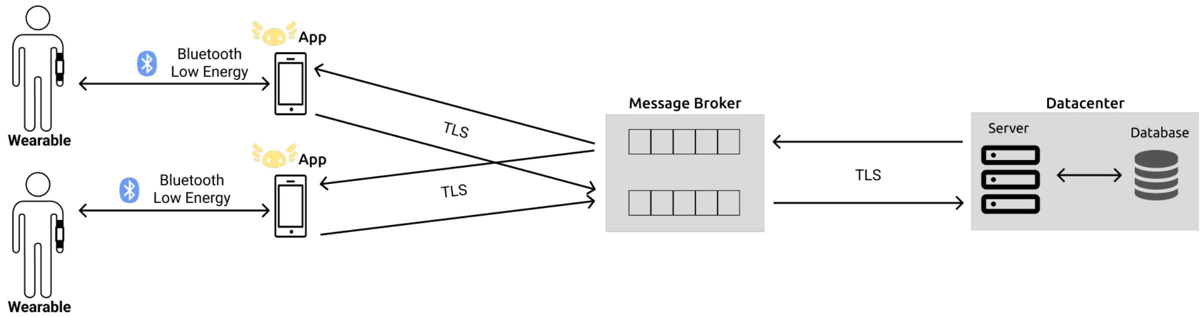

Figure 3.1: Anxolotl System Overview

The message broker, which serves the main purposes of uncoupling the data generator from the data consumer, and adding some fault tolerance, then receives the data. Our message broker will have to use the publisher subscriber model, to allow for this decoupling, and also guarantee storage in the case no one is there to receive the data. Furthermore, if everything goes as planned, the data will be retrieved by the datacenter via TLS as well.

Lastly, the datacenter will focus on generating value from the captured data, to this end, we will be using our ML engines, which will be performing the work directly on the server. They will not be trained there, but simply ran there as a stored model, to lessen their impact on the server, which will still be high. The Engine's results will be returned *via* TLS to the message broker, so that data can be consumed by the app, whenever the app is able to do so. Another relevant feature of our datacenter, is the storage of data, for further investigation. The stored data in the datacenter is anonymous, so only the user can see their own data in their own mobile app.

3.2 Implementation of the Anxolotl System

This section will be divided in three parts. The first one, regarding the Anxolotl App, where we will deeply analyse its implementation, in an effort to present all the features a user can expect to achieve while using our app. Then, the Message Broker section, in which we will focus on our implemented way of sending messages between the Anxolotl App and the Datacenter. Lastly, the Datacenter, which performs the main task of executing the heavy lifting by calculating features and applying them in pre-trained models to return the results.

Figure 3.2: Maxim HBand SP-2D-M

3.2.1 Anxolotl App

Making a good first impression, is certainly the way to start an experience with the right foot. Since our application hosts the point where a user will contact with our system for the first time, it must be easy to use, not crowded with options and relatively good on the eyes. If those are the reasons to start, the features we provide, and the ease of continued usage, are the reason to keep using it.

In this section we will go over the main points of our app. We will start by addressing the wearable, which is considered part of the app, as well as its strong and weak points. After that, we will present the implemented mobile application divided into two parts, the technical behind the scenes part, where the workflows lies. Some of these technical aspects include authentication, how the data is handled throughout or app and storage solutions. Finally, we will present our app's presentation view and its main user features, with some theoretical support when needed.

Wearable Initially, we wanted our solution to provide a generic entry point to all the supporting BLE wearable devices. That was harder than expected, and so we had to focus on a single device to run our prototype in a seamless manner. Our device of choice was the Maxim HBand SP-2D-M, shown in Figure 3.2. In this section we will go over the important aspects of this wearable that makes it suitable to our system.

The first mentioned part in this thesis, that we exacerbated as our main ask for a band was the sensors, at first we thought we would need sensors such as EDA and BR, but in the end a simple PPG coupled with an ACC that can detect and identify activities was enough. While these sensors are not uncommon, the sensors this band provides are accurate, even if in the connection department they present some hurdles.

Another important part is the battery duration, and while this band is advertised as having a 4-5 day battery life in discrete mode, we are not sure how well does this translate to real life. Having at least 2 and half days of battery is quite important, to avoid making the users have another daily charging device, which could undermine the project viability.

This band has the possibility to program and communicate via an exposed Application Programming Interface (API), which allows retrieving data as a stream with notifications, allows to change the measurement mode from continuous to discrete, and presents more customizable parameters. The most important features in this regard is the implementation of standardised open Bluetooth Consortium services such as the Battery Service and the Heart Activity Service. On the less positive side, the activity service is proprietary [39] and its implementation can not be used with other wearables.

The ability to change sampling rates in the discrete mode is also quite important for this project, even though the band does not totally support it. But, another contender for a supported wearable would have to support it at least as good as this band. The important data coming from this band ends up being the activity type, the workout detection and the heart rate measurement, which is accompanied by the IBI, but that could have been calculated from the HR, since it is instantaneous, and each is the inverse of the other.

Lastly, any potential wearable that could be added to our solution would have to implement Bluetooth Consortium services of Heart Activity, Battery Service and Physical Activity Monitor to be compatible with our application. It would also need to be connected using BLE.

Mobile app The behind the scenes aspect of our app was elusive at best, since we did not know what to expect from this implementation. We had worked with BLE connectivity before, but have never implemented a system that used constant background work. So this was poised to be the most challenging aspect of our implementation. Our starting point in this section is going to be the technology decision.

Our intended path for the app implementation is to use the Flutter framework together with the Dart programming language. This decision has multiple reasons, but the main ones are the support to iOS and Android systems out of the box and the ease of implementation with a big library of supported plugins. The group is also already familiar with Flutter and Bluetooth connections in it, which played a role in the choosing of this framework. Being a highly performing language and widely supported is also attractive from the point of view of future support. Lastly, the object-oriented nature of this language, while being statically typed, but allowing a dynamic type, makes this language just elastic enough for a project of this nature.

From the beginning, our app was intended to work with most common mobile operative systems, namely iOS and Android, and even though we are not able to test on iOS due to technological deficiencies, our development will be taking it into account. By this, we mean that every plugin, method and development is done by checking if Flutter supports it on iOS as well as Android.

The app contains multiple parts, which range from more software and algorithm based to parts that are more interactive and user facing. We will start our App overview by going over the software based parts, and ending in the user facing features. All the app screens can be found in Appendix A.

Authentication of the mobile app Authentication (Auth) was not always part of the equation throughout the development of the app, but the need for safety made it almost impossible to go on without some form of authentication. Anyone with access to our app would be able to reverse engineer it and get the broker data and send messages. This was a problem, and so we needed to develop a robust yet lightweight authentication solution.

To this end, at first we turned to Google Auth and Apple Auth, since those would be our target systems for this app, and a user would necessarily have a Google email or an Apple Sign-In. This worked to some degree, meaning that Google worked, but the Apple solution was near impossible, given their strict requirements, technological and monetary, so we had to come around with a solution that could serve both our user bases (iOS and Android).

We found Auth0 [51], which is a flexible, and almost drag and drop solution to add authentication and authorisation services to multiple applications. It uses the OAuth 2.0 protocol and provides great customisation and transparency in the way the data is stored. Since we only needed to store an e-mail and a password, and then get an ID token back, this solution seemed promising. A Flutter plugin is provided for the Auth0 implementation, which we used to make our integration seamless, and that is one of our ways of authentication, and the only one that supports both platforms, while asking for minimal data.

As stated before, the other authentication method available is Google Auth, which also relies on OAuth 2.0, making it on par with Auth0. The user can simply pick any Gmail account present on their Android device and enter our application. We just use the email and ID token data, so no sensitive data is used from there.

Both platforms support the problems of a user forgetting a password, needing to log out, not remembering e-mail and any other common issues. This decision was taken to provide a solid authentication interface with minimal effort, which would be hard to implement and would need a lot of time. In this we case we simply choose to prioritize the nuclear aspects of our thesis, such as the algorithms' development.

Storage From the beginning, we wanted safety and anonymity in the physiological data storage, and safety regarding storing all the personal data. The data calculated by our algorithms can be stored in the phone indefinitely, since it has no real value besides helping the user monitor their anxiety levels and adjust accordingly. Contrastingly, the physiological data could be used with malicious purposes and so, it should be stored safely.

Our app captures some physiological data related to the person, some of each is mandatory such as gender and age, and some is optional such as weight and height. All this data is stored per user in Flutter Secure Storage (FSStorage). This storage is used for small data tuples in the form of a String and Value. This secure storage is encrypted using Keychain in iOS, which does not have the implementation details public, and Advanced Encryption Standard (AES) encryption in Android. In Android, the secret AES key is encrypted with an Rivest–Shamir–Adleman (RSA) key, which is then stored in Android's KeyStore. The Bluetooth device information (for connecting automatically on start), is also stored here.

The data received from the Datacenter is also stored, but in that case, it is stored in a NoSQL database named ObjectBox. This solution provides a very reliable way of storing big amounts of data fast, while being easy to implement, maintain and query. Here, we store the daily and hourly anxiety levels, as well as the panic values. In a last case scenario, we also use this database to store temporarily the physiological data, until a synchronisation happens, and then the data is deleted from the database.

Connectivity The name of the game is connectivity, meaning that our app heavily relies on it, since we will need internet access to upload files, and Bluetooth access to receive the data and communicate with the wearable.

Our internet connectivity status is always checked before using, to avoid exceptions that can stop the app normal usage. In the case of uploading data to the Broker, our app will first check if there is connectivity and if not it will simply keep the data stored until Internet access is present. Our app can work with both mobile data and wireless data.

We also use Bluetooth to allow connections to the wearable. Most wearables nowadays run on some version of BLE, and that is the type of connection that we have the more developed support, but some older wearables still run on Bluetooth Classic. We also support Bluetooth Classic in our app, for any future endeavours, but currently the focus is to present a solid BLE foundation in our app, to work seamlessly with the band we intend to use.

Data Collection The data flow is one of the defining characteristics of this application, namely in the way that our flows work. While a simple blocking workflow would work, it would present a bad user experience, given the user would not be able to access the app while the data collection was happening. For this reason we had to rely on a notification flow to perform most of our app's work.

The data flow starts when the user tries to connect to the wearable. At this point, if the user has not entered their mandatory health attributes, which are age and gender, they will be asked to do so, since this information is vital our calculations. If the user forgot to turn on Bluetooth, they will also be denied to access the scan menu, and be asked to turn it on. In the case everything is alright, the user will be taken to the search Bluetooth Devices page.

In this page, the user will be able to pick a device to connect to. If the connection is successful, our wearable workflow will start, as shown in Figure 3.3. Upon a successful connection, the app will immediately store the Bluetooth device information in the FSStorage, and if this operation is successful, the app will retrieve the battery information of the wearable to show it to the user, and disconnect the band in the main thread (UI thread).

After this, it is time for our notification workflow, which will be explained later. Eventually the notification workflow ends, usually when the user stops it by forcing a disconnection, or in the case of a fatal error. In both cases the notification will disappear, and the user will know. But in that case, our app will disconnect the device in the notification thread and remove it from the FSStorage, before ending the notification, so our app knows that a device is no longer connected to it. Upon inspection of the device information page by the user, the app will query information from the FSStorage, and return no device connected.

Regarding the notification workflow, it is a foreground service, since a background service notification would need the app continuously running in order to work. Contrastingly, a foreground service is free to run autonomously from the app that started it, to the point that our notification will run alone if the smartphone is rebooted, in the

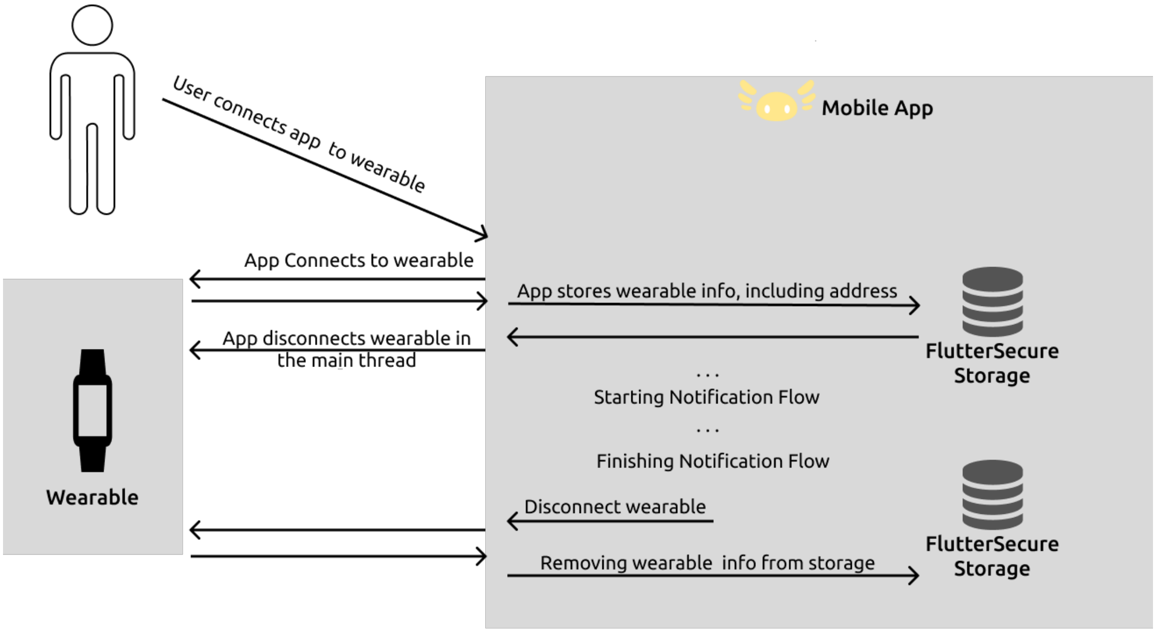

Figure 3.3: Wearable workflow

case a stored device exists in the FSStorage database. The capturing, only ever stops when the user intends to. On the not so bright side, a Foreground service does not have access to any resources used by the main thread, and that is the reason we have to disconnect the device in the main thread, to be able to connect in the notification work.

First things first, our work starts by loading the necessary resources from the main thread, namely the device information, since FSStorage is one of the few resources that can be shared across threads. This will not generate concurrency problems, given the notification thread will only access FSStorage when it has a connected device, while the main thread will access it when there is no connected device. After this, our work notification will load all the necessary components to run, namely the user id, and the measurement configurations such as sampling rate, and sampling mode (Continuous or Discrete). Finally, we set up a Cancellation Token that will be used later. We also load the Bluetooth device information, and connect to the device, to set up all the necessary components. All of these and the next processes can be seen in the simplified notification workflow code presented in Listing 3.1.

After the configuration, we initialise a listener, to listen if the user from the main thread intends to disconnect the service and inherently terminate the notification. In the case of the user terminating the notification, we set the Cancellation Token to true, which is passed to every single measurement, and in turn, every still ongoing measurement will stop, independently of their status. If they have the necessary data, the results will still be stored, if not they will be discarded. After that, we check if the Bluetooth device is connected, and if it is, the app disconnects it, and proceeds to remove its information from storage, so the main thread will see that no Bluetooth device is present. Lastly, we stop the notification, and after that, only the main thread is left running.

```
notification = Notification()
userId, measurementConf = loadConfigs()
cancelToken = CancellationToken()
bluetoothDev = Storage.loadDevice()
connectToDevice(bluetoothDev)
service.on("stopNotification").listen(event {
  cancelToken=true
  if (bluetoothDev.isConnected == true)
    bluetoothDev.disconnect()
  Storage.removeDevice(bluetoothDev)
  notification.stop()
  })
# Work Function
Timer.periodic(Duration(seconds=30), () {
    if (bluetoothDev.isConnected == false){
      bluetoothDev.connect()
      if (bluetoothDev.isConnected == false)
        throw Exception ("Device not found")
    }
    measurement = startMeasurement(measurementConf, cancelToken)
    Storage.storeLocal (measurement)
    notification.update("Success")
# Synchronization Function
Timer.periodic(Duration(minutes=20), () {
    if (NetworkValidator.isNetworkAvailableConnected()){
      measurements = Storage.readLocal()
      sendMeasurementsToBroker(measurements)
      Storage.removeLocal(measurements)
    }else {
      throw Exception ("No internet available")
    }
```

Listing 3.1: Notification process in Dart

If the user starts a connecting and it is successful, our notification starts two periodic Futures, one which will be tasked with doing the capturing and processing work, and another one tasked with synchronising the data with the message broker. They have different periodic intervals, the work functional interval refers to how much time passes between the repeating each measurement, while the synchronisation interval pertains to the interval between trying to upload the local stored data to the broker. Regarding the specific intervals they are 30 seconds for the work function, mirroring what is used in the Anxiety Engine, and 20 minutes for the synchronisation function, which allows the measures not to be too outdated in the case of errors while also not spending too much time using bandwidth.

Starting with the data work function, firstly, it checks if the Bluetooth device is connected, and if it is not connected, tries to connect it to the smartphone. If this also fails, the notification content is updated to inform the user that the device could not be found, and we expect the user to fix that. Whenever the user fixes the problem, the app will be able to perform the work function as expected.

If the Bluetooth device is connected, we start the measurement, and take one discrete sample of both IBI and activity. The sample will be filtered during sampling, namely we will check if the band is being worn, and if the band is not being used, every taken measurement will be discarded, until one passes the filtering. If twice the duration of the sampling interval has passed (1 minute), the app will procceed to discard the measurement if it contains no suitable samples, in an effort to not clog the volatile memory.

Our synchronisation function is quite simple, when compared to the work function, since it just performs a check, and it runs the function. The verification is about whether the smartphone has connection to the internet or not, and if not, the notification is updated with a warning to the user that no internet connection is available, and the function ends. If there is connection, the function retrieves all stored measurements in the local storage, and uploads them to the broker, and then deletes them from the local storage. If there is a failure during the update, the data which was successfully uploaded is deleted from storage, while the rest stays in the local storage.

With our work and synchronisation function, we end the part regarding the main processes of our application. There are more implemented processes, but their importance is not as big to the whole flow of the application, and these are the nuclear aspects of our implementation. After this, we will explain how the work from all of these processes is shown to the user.

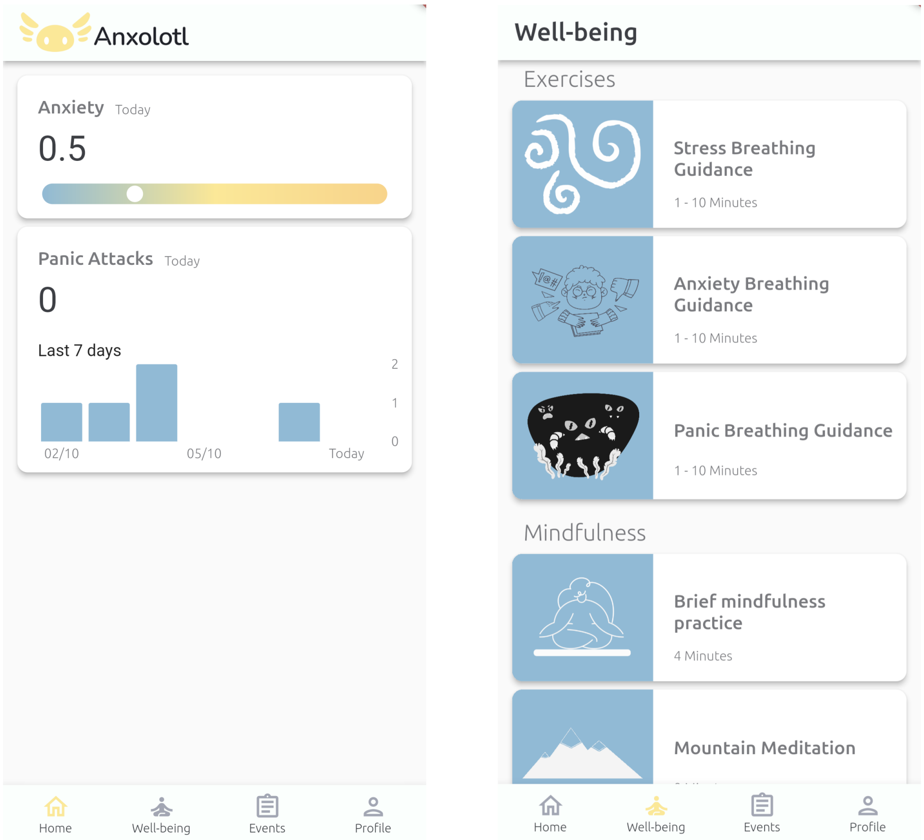

Figure 3.4: Anxolotl app dashboard (left) and wellbeing (right) section

Dashboard The heart of the Anxolotl App is the mobile app itself, which is developed as the crown jewel of our system. While having a section about design and the user experience does not suit the theme of this thesis, we can make a small detour about it, since it is important to note that an effort is going to be made on that aspect. But pertaining to this thesis, some of our biggest contributions are the main features we are able to craft and present to our users. The app is focused on ease of use, and a very hands-off experience, while we tried to add features for the user to use whenever they want to, the app is also suitable to be used without much involvement. We will present this paragraph around our most pertaining features, which will be the main menu pages showing the possibilities, and the notification that signifies the user what is happening on the background.

Our color choices rely on some subjective logic, but have a reason to be. The three primary colours that were dragged along all the development of this thesis, and are also an integral part of the application design. They are pastel yellow, sky blue and afternoon orange. Both the yellow and orange were chosen for their association with happiness, cheerfulness and light-heartedness, the paler versions were used to grab a calmer, more soothing tone. The blue was used for its strong ties with mental health and for being a calm colour as well.

In regards to the features, whenever a logged-in user enters our app, they are presented to this screen. Here they can see every aspect of their monitored aspects, be it anxiety levels or panic attacks. The first menu as presented in Figure 3.4, is a quick overview of the user's current status.

If the users click on the anxiety card, they will be taken to another screen, in where they will be shown a more detailed view about that health statistic. In the case of GAD, that will be a current level, the monthly levels until today, and the highest level in the day that the data is being shown. We have particular emphasis on the top portion, where we will show the daily average anxiety level, which we use as the most useful marker.

On the dashboard we also have an overview of the PAs statistics, which are not as numerous as the anxiety statistics, and so we believed there was no need to implement a detailed view. The presented data pertains to just 7 days, in an effort to show quick improvement to the user, instead of worrying them with long time gone events.

Wellbeing exercises Another big feature of our app is the ability to present solutions to high levels of anxiety, or the detection of a PA. For this purpose, we devised the wellbeing area in the app, which is presented in Figure 3.4. In this area, we present all the supported help the users can perform by themselves, which are breathing and meditation exercises.

Our breathing exercises are presented in an animation form, with a time to inhale, hold and exhale, which the user of the app should be able to mirror. These limits are well demarcated within the application. For all our breathing exercises we took inspiration in the available literature to present the best help exercises that were implementable by our team. To this end, we researched thoroughly all the presented breathing exercises.

Regarding the stress exercise, stress is strongly tied to the hormone cortisol, so in this case, our biggest inspiration was the paper from Ma et al. [27]. There, the focus centres on diaphragmatic breathing and its potential to improve cognitive performance and reduce negative subjective and physiological consequences of stress in healthy adults. They used a four breath per minute average, and their results were great. Regarding the timing between inhaling, holding an exhaling in seconds, we took inspiration from [37], which uses with some success, a 4:0:6 breathing technique for stress management. Since we need 15 seconds per respiration cycle, we are using a 4:5:6 deep breathing, in order to respect the 4 breath per minute orientation for cortisol reduction mentioned in the first study.

A study from Lin et al. [26] revealed that 5.5 breath rate per minute, coupled with an inspiration to expiration ratio of 5:5, had the highest increase in vagal PNS activation than other breathing patterns. This result can be applied to clinical populations with a low HRV, such as coronary artery disease, major depressive disorder, and anxiety disorder [26]. It was also referenced that a 5.5 breath per minute rate offered the best advantages in lowering LF/HF ratio, so we devised a breathing plan of 5:2:5 for our anxiety breathing exercise, based on this paper.

Lastly, our panic breathing exercise inhale:hold:exhale ratio was strongly influenced by Song et al. [48]. Panic attacks cause multiple symptoms, and can cause a decrease in HRV as well as an increase in LF/HR ratio when comparing to normal values. This study, focused on multiple things, but one conclusion, was that a respiratory breathing rate of ten breaths per minute, yielded the lowest LF/HR ratio. We used that information and devised the panic breathing exercise in a 3:0:3 manner, with equal inspiration and expiration time, to ease the exercise. Since a person nearing the occurrence of a panic attack needs help fast, it is far better to provide a simple and easy to perform breathing technique.

The second part of our wellbeing exercises pertains meditation. And, it is widely known the effect of meditations on the human psyche. While the presented exercises are not oriented to a particular scenario, they are here presented as a resource to be used at the user's discretion. We currently offer two meditation exercises, a short one – "Brief Mindfulness Practice" - and a longer one – "Mountain Meditation". These exercises were taken from a free open source repository named Free Mindfulness [11]. The only current available meditations are in English.

The meditations are played with a music player, and the user can choose to start and stop them, in the case they are interrupted for some reason. They can also go back or go forward on the part of the meditation they are using. In the end a final menu is presented congratulating them for a successful meditation.

Events Section The Events section is quite simple, since its main purpose is to warn the user about particularly high or uncomfortable health values. While it might not always translate to symptoms on the time of showing an extreme event, it might translate in the future. In this case the users are tasked with reacting appropriately to the presented information.

Figure 3.5 illustrates all the possible events our app is producing. In the case of a panic attack, the user will probably notice it, before seeing the app, but the other cases might cause the user to accept our app's help and feel better than before.

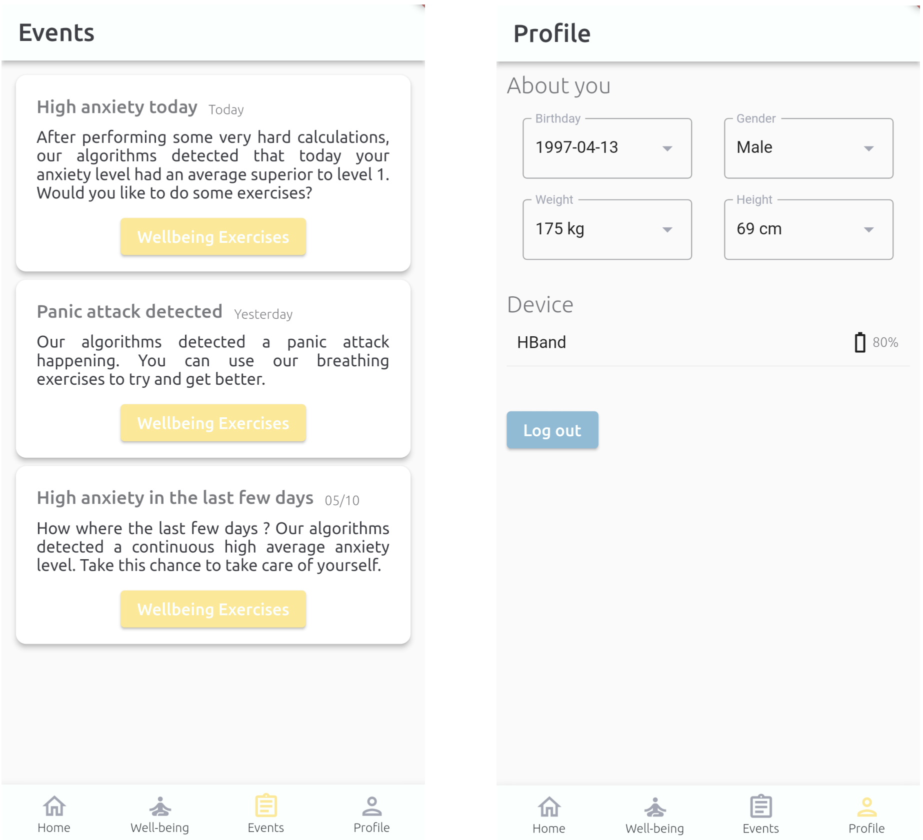

Figure 3.5: Anxolotl app events (left) and profile (right) section

All our events have a call to action to perform wellbeing exercises, in an effort for the users to quickly address our concerns with minimal effort. The point of this section is mainly informative about outlier cases, such as a panic attack, high anxiety during a day, which translates to a daily average bigger than level 1.5, or high anxiety during multiple days, which means an anxiety daily average level higher than 1 during four consecutive days.

Profile Section A user that just started using the app will be taken to this section, in which they will have to insert their health data as well as their intended device to connect. This section relies on being a central hub for a user's data and settings. Currently, we do not offer any settings, but on a future iteration, this would be where they would fit in.

As presented in Figure 3.5, we have multiple health characteristics. Firstly we have age and gender, and as stated before, they are mandatory, and will lock the user out of using the rest of the app, if they do not insert them. We also have the height and weight, which are optional, and more focused to record data for future research on our algorithms.

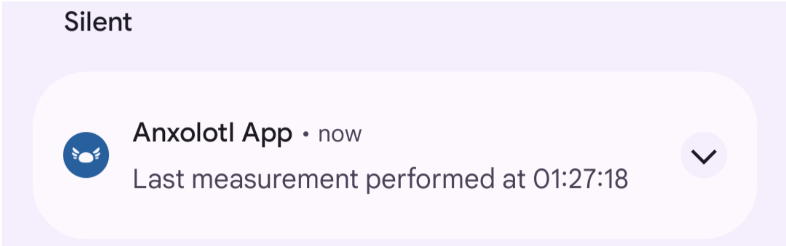

Figure 3.6: Anxolotl App measurement notification

Lastly we present the Bluetooth device connector, that upon click will take the users to another page where they can see which devices are nearby, and can pick one device from there. There is some filtering in the devices show, particularly, we do not show unnamed devices since all the wearables the group has contacted present the name of their manufacturer at least and this minor feature improves the user experience.

Notification While a notification usually is nothing to write home about, in the case of the Anxolotl App, it is intended to at times be the only point of contact between the user and the app. Its main purpose is to inform the user that the app is working correctly under the hood, without being intrusive.

Our notification was made as a silent notification, so it does not pop on screen, and it does not make noise, almost as a presence light just being there. It presents the user with relevant information regarding the measurements, such as when was the last measurement was performed or if the last synchronisation happened without issues. In the Figure 3.6 it is presented a successful measurement.

This notification is also here to inform the user if the device could not be found upon a measurement, so the user can take action regarding that, such as bringing the device closer. On the case of synchronisation, it will warn the user that connectivity is off, and so the user should turn it on, so the next synchronisation does not fail.

Lastly, the notification existing is a proof that the notification flow is running smoothly in the foreground, since whenever the notification is up, the notification flow is running. If the notification disappears, it means a fatal error has happened, and the user should start the app and probably reconnect the device.

3.2.2 Message broker

During the development of the system, it was intended for it to communicate via Hypertext Transfer Protocol Secure (HTTPS), but after some thought about this issue, it was noted a system using multiple smartphones always sending data would scale badly with that pattern. It was noted that our system is closer to an Internet of Things (IoT) system than a typical server client architecture.

Given these similarities, at first we thought an Message Queuing Telemetry Transport (MQTT) solution would be the best suited approach to our system. We implemented and tested Mosquitto MQTT, and it even provided safety features, but we needed to keep the messages in the queue while they could not be processed by the datacenter, and they needed to be kept there in the case more messages arrived. Mosquitto MQTT could not provide this scenario, since whenever a message entered, the last one was erased, and so we thought Publisher Subscriber Model (pub sub model). A pub sub model solution would allow us to achieve of some of our intended values, such as scalability and modularity. A solution of this calibre would allow us to decouple smartphone users and the datacenter providing decoupling in the system. It would also be suited to modularity, since we could create multiple datacenters for different types of data, i.e. splitting between GAD datacenter and PD datacenter.

It would also add some fault tolerance, since we could have the datacenter down, but the apps could still continue working as normal and no data would be lost. Regarding future proofing, this solution also allows to migrate the datacenters without turning off the whole system, and all of these points made this solution very attractive to implement in our system.

We have chosen to go with the RabbitMQ solution, for its long history and reliability over such time. It is a fast and very tunable solution, and the amount of customisation will be important in delivering messages to specific users, instead of relying solely on the app to do the filtering work. It also provides some interesting security options, regarding TLS, that we will explore later.

Security The message broker is our security hub regarding the communication. It is here where we implement our connection security measures, and where we store our certificates. By being an isolated piece in the puzzle, it also allows us to keep some isolation from the datacenter, where our anonym health data is stored.

Implementing TLS was quite simple, all we had to do was to create a self-signed certificate, which issued certificates that are used in the datacenter and in the Anxolotl App. These certificates guarantee that whoever has access to our broker is authenticated as a trusted entity, and can use the system.

Every entity has a different certificate, for the case of detecting a compromise in our security and need to void any certificate from our root certificate. In that case an outdated version of the application for instance would be out of access to our broker and could not be used any more in the system, until an update. TLS also provides secure communication, *via* a cryptographic protocol, and so our communications will be less subjected to attacks of man in the middle, which could undermine our data safety. While not a solution for every attack imaginable, it provides our system with a good amount of security for the foreseeable future.

Broker Architecture Using the Message Broker was one solution found to provide scalability while adding some fault tolerance. The latter was achieved *via* the existence of the broker itself, given that, in the case of the datacenter or Anxolotl App sending messages while the receiver is down, will result in them being stored in the broker. On the scalability side, we intended to add an easy way of adding more datacenters, in the case of future expansion.

Firstly, we will reiterate the used Broker exchange typologies. In this system, we use two of them, the Fanout typology and the Topic typology, but there are four in total. The Fanout typology sends the messages to all the subscribing queues, while the Topic typology sends messages to the queues that have a routing key corresponding to the message. The messages also have properties that might be used to differentiate between them. In this case, we use content-types from the message properties in multiple occasions.

This explanation and reasoning will start from the place where our data is sent, the Anxolotl App. Here, we are tasked with sending three important informations, the user's health data and Bluetooth device, and the wearable captured data. Both the health data and Bluetooth device data are sent through the user_info exchange, and are differentiated *via* content types. This exchange uses a Fanout typology, as shown in Figure 3.7, to allow for further expansion, and creating a new queue for a new datacenter that might be needed, in this case the Fanout typology is unnecessary, if we ignored the future expansion point.

Next, we have the hrv_data exchange, where the data transfer also happens from the mobile app to the datacenter. We apply once again the Fanout typology, to ensure that further developments, which might include creating a new queue for stress, are supported. With this typology, we guarantee that every queue receives the physiological data, and in this case we are just using two queues with the same data, to further down the line ease the split of our datacenter into two. These two would be tasked one with `hrv_anxiety` and the other with `hrv_panic`.

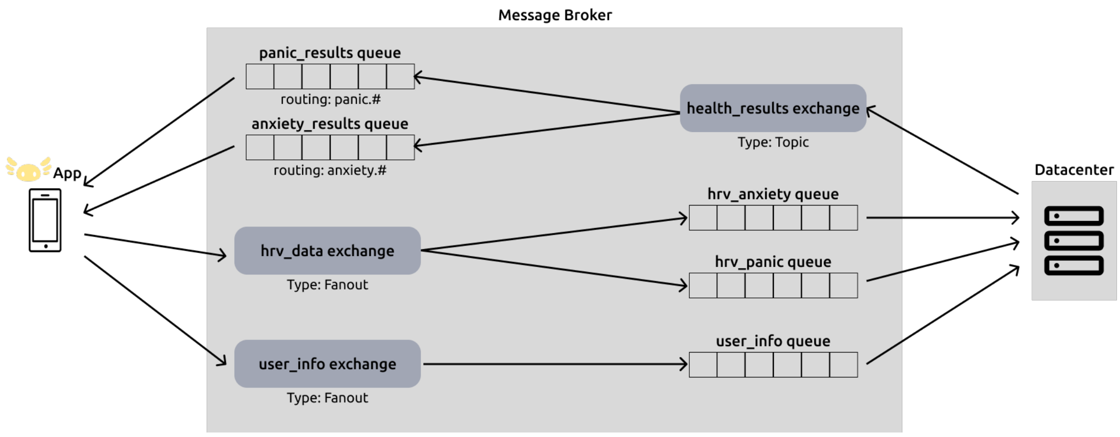

Figure 3.7: Message Broker exchange and queue architecture

Lastly, we have the health_results exchange, in which the datacenter sends the calculated labels to the mobile app. In this case, we are using routing keys to differentiate between queues, which means the data from the `panic_results` queue will be totally different from the results in the `anxiety_results` queue. We also added further granularity, by citing the type of measurement after the intent such as in "anxiety.hourly", which refers to a label of one hour, pertaining to anxiety level calculations.

3.2.3 Datacenter

Being in the shadows usually is associated with a bad thing, but in the case of our datacenter it is not. While not as prominent as the Anxolotl App, it could be defined as the brains of our system. It is here were the physiologic data is processed and transformed into valuable information about the user's health status.

Our datacenter is composed by two components, which are the server and the database. The server does all the processing, while the database is used to store the anonymous physiological data for future analysis and algorithm development. This section is divided into these two components, and we will present some lights about their development and functioning.

Server Initially intended as a part of the client-server architecture, our server maintained its name. While no longer abiding by that architecture, since our system bears more resemblance to a distributed system, its functions remain largely the same. Our server is tasked with receiving the data from the broker, processing it and returning the results to the broker, so they can be seen by the correct user.

Our server is implemented in Python, and that choice was made with speed of development and compatibility with the Engines in mind. With more time in our hands, the group would have probably optimised the datacenter to Kotlin or Java, but as a proof of concept, a Python server aids a fast development. With this help, it was also possible to spend more time researching the algorithms, instead of developing an extremely optimised server.

Our server connection starts after the message broker is configured, by loading the pre-trained ML models onto itself, connecting to the database and lastly connecting to the broker. After this connection, our server will subscribe to all the queues related to receiving physiological data from the app. Lastly, the server will be listening to its subscriptions, and upon the arrival of new data, will start its flow.

When arriving in our server, the data is immediately stored in the database *as-is*, in an effort to not taint the data that might be used for further analysis in the future. After that point, the data is filtered using similar mechanisms to the ones used to filter data in the Engines.

In the case of the anxiety level detection and panic detection, we discard the IBI data which was captured during exercise, due to a higher probability of noise and bad readings. After this first filtering we apply the same IBI outlier detection, by removing the values below 180 ms and above 2200 ms. After this simple filtering, we use the filtered data to calculate the same features we used in the Anxiety Engine and feed them to the model. The Panic engine behaves similarly, with the only difference between the two being the features used by each.

Regarding the filtered data, after performing the filtering, we then proceed to store the data in the database. Since we receive data at 15 minutes intervals, but perform calculations that take into account hourly values, such as anxiety, we would have to re-filter the data from 45 minutes every time we were trying to calculate hourly anxiety. This optimisation is intended to ease the burden on the datacenter, since like this, we apply the filtering on a given entry of data only once. We do not store features, since the calculated features have always different data, and so, there would not be a point to storing it besides future-proofing, since currently we are not re-using any features.

After getting the classified labels from the model results, the server sends them back to the broker, with the specific type of data identified, as well as the user they are intended to. Our server calculates three types of data, hourly levels, last hour levels and 15-minute levels, the first two being used for anxiety and the last one for panic. The last hour levels are always calculated whenever new data arrives, and are intended to be discarded after a new calculation. The hourly levels are intended to be kept, and are calculated each time the minutes reach zero in real time, it is used to calculate daily and monthly averages in the smartphone end, and it only applies to anxiety, since the panic state is used as a counter, given its binary nature.

Regarding the performed work, we have multi-threading, with various threads doing different work. The threads we have working are:

- User Info;
- Anxiety Queue;
- Panic Queue; and
- Work thread.

Since, the data we use is the same in Anxiety queue and Panic queue, we opted to use a simple thread to perform the work and avoid redundancy in a single datacenter instance. We use the Panic Queue thread purely to acknowledge the messages, but no work is done with that data.

The User Info thread is the simplest of the four, as it listens to the queue of the same name. Whenever a new message is received, it performs the distinction of the data as either user's health information (Birthday, Gender, Height and Weight) or user's Bluetooth device(address). After creating a Data Transfer Object (DTO) for the type of data, it stores said data in the database, and waits for the next message.

While not exactly equal, the Anxiety Queue thread performs similar work, it listens to the physiological data, which can be physical activity or HRV. In the case of the physical activity, the data is simply transformed into a DTO and stored in the database. The case for HRV is identical, but in the end, an entry is added to a concurrent Map, identifying the user and the first timestamp of the received data.

This concurrent map is accessed by two threads, the Anxiety Queue thread and the Work thread, and so it is protected by a lock, since we have operations using reads, and deletes based on the reads. It is the only piece of shared concurrent data we have in our server code. This map is used to identify to the Work thread which users have data waiting for results.

Lastly, the Work thread performs all the filtering, and calculations already referred above, and ends up publishing both the Anxiety and Panic labels to the broker. It also accesses the database tables, only to perform reads, and writes only in the filtered data table, which is not used by the other threads. The way to deal with the map, is by locking it, copying the users from it and clearing it, and lastly releasing the lock, so no thread is using the map while the Work thread is changing it. This allows the server to receive new data from users while work is being done.

This threading system was the way the group found to receive multiple data from multiple users, while not blocking users too much. The threading could be more granular, and more precise, allowing for even more optimization, but the datacenter's ability to always receive data was the main purpose behind this implementation. The work is also calculated on multiple users at a time, although on a series manner.

Database Initially, a database did not make sense, given that we could simply calculate our data and leave it in the broker, while it was not processed. Upon realising that the public datasets for GAD and PD research are almost non-existent, we realise that we have the opportunity of increasing our knowledge base. With this opportunity, the group understood there was value in storing said physiological data for future analysis.

The datacenter database is coupled with a datacenter, in an effort to avoid a centralised database, that would crash in the case of failure. If we were to have multiple datacenters, we would also have multiple databases, that would not be synchronised. Probably they would contain the same data, since they share the source which is the message broker. In any case, failures could happen which would lead to different data in the databases, but that would not be a problem, since those errors would be within an acceptable margin of error for a research purpose.

Our database is implemented using PostgreSQL, a SQL database, which offers security and a relational database system. The choice of using an SQL database in this case was to accommodate concurrent accesses from multiple server accesses. Our transactions always carry a level of isolation, to allow for this concurrency aspect of the datacenter, which could happen, since the datacenter work is potentially slow, given its computational complexity. Lastly, this SQL solution was chosen due to its compatibility with exporting the data in a ML friendly format such as Comma-Separated Values (CSV), which is supported out of the box by this solution.

4 Methodology

The operation of devising a new algorithm is a process in on itself, while not being a massive undertaking, it includes a lot of iterations through trial and error. For this reason, we have this chapter devoted to explaining the entirety of the devised algorithms in this thesis project. We are calling our algorithms by the name of engines, since the amount of operations done by them is big. The task of receiving the data until generating new values is a whole pipeline, and so calling it simply algorithm would undersell the idea.

This chapter is divided into sections regarding each engine's purpose, and so we have Anxiety and Panic Attack. Each of these sections will have multiple subsections, regarding dataset manipulation, data normalisation, until the feature creation, and lastly we will explain each algorithm.

4.1 Anxiety Engine

In this section, we are trying to sum up all the work performed to create a viable real time anxiety level classifier. It is not an easy feat, since from the start, the public datasets are almost non-existent, and the amount of research on the topic of anxiety real time detection is small and not always open regarding their algorithms. We intend to present a starting point for someone to grab and further develop, and so all the processes and developments regarding our anxiety classifier, or Anxiety Engine, are presented here.

4.1.1 Datasets

Regarding the monitoring of anxiety, the existence of public available datasets is quite reduced, and so we had to adapt to those conditions. After performing an exhaustive search on this topic, we found out two datasets that could be interesting for our algorithm.

The first dataset is the Multilevel Monitoring of Activity and Sleep in Healthy (MMASH) dataset [43], which analyses the data from 22 healthy young adult males with the duration of 2 days. Most of the physical characteristics of the participants were recorded such as height, weight and age. The participants took some questionnaires as part of the onboarding regarding their mental health, such as morning-readiness tests and daily-stress levels regarding the last 24 hours before the experiment. This experiment consisted on monitoring the everyday lives of these individuals while monitoring their health data.

The participant's data was recorded using a Polar H7 monitor to record HR and HRV data and an actigraph to record physical activity, sleep quality and ACC data. Daily, each participant also took a State-Trait Anxiety Inventory questionnaire (STAI), and took a questionnaire to assess their emotions with an interval of 4 hours between each questionnaire. Their saliva was also tested during the morning and before bedtime to detect cortisol, a hormone highly correlated to stress levels.

This dataset is a good fit for our project, since it is able to target the monitorisation of daily anxiety, with data that we will also use such as HR, HRV and activity data. Another very good point about it, is that is labelled with an anxiety questionnaire which was taken daily, fitting perfectly into our ambitions. But this dataset has some big drawbacks as well, such as not being representative at all, given the participants are all males, and they are young adults as well, and both age information and age information have been shown to have a big correlation to anxiety disorders. Another drawback is the size of the dataset, since even though we are able to double it from 22 participants to 44 entries, since two days were measured with two labels, it still is a strikingly small dataset making it particularly prone to overfitting.

Another dataset that caught our eye, is the Anxiety Phases Dataset (APD) [46], which analysed 95 young adults during two anxiety experiments, which are a social anxiety experiment and a bug phobia anxiety experiment. The public available dataset is a subset of the real one, with just 40 participants per experiment. Beforehand, each participant filled questionnaires regarding their usual anxiety, stress and depression levels. They also rated the anxiety and avoidance score for each of the tasks. Each participant also presented their age, gender and ethnicity, and regarding gender, this dataset provided 51 females and 44 males. It is also important to note that this dataset was recorded in a lab setting, and so the sensor's accuracy should present better results than our solution's data capturing.

This dataset's *modus operandi* was to present each participant with a stressor. It was known beforehand if each participant had a fear or phobia to that particular stressor, and so they were divided into high anxiety and low anxiety groups for each task. The experiment on itself included various stages, the baseline stage, the instructions stage, in which the participants were informed about what was going to happen, potentially increasing anxiety, the exposure stage, and the relax stage. It was also recorded each participant's reaction, which was categorised as safety, avoid, escape and confront. Another important detail is that in each of the experiment stages a Subjective Units of Distress Scale (SUDS) questionnaire was taken, which was helpful to determine how much distress the participants were being affected by at each stage. The SUDS questionnaire deals with a distress scale, which depending on the context can be attributed to fear, stress or anxiety, although it is mostly used in anxiety measurements.

During the data capture, each participant had multiple sensors, such as ECG, EDA, torso posture, and activity from the wrist and leg. While this lab setting might conduct to higher accuracy, the amount of sensors might also change the natural responses.

Regarding the application of this dataset in our project, while not as perfect fit as the MMASH dataset, it still provides good value. The experimentation with a more representative group, albeit small, brings value to the classifier, and the existence of repetition in the experiments with both high anxiety and low anxiety groups, allows the isolation of variables. On the negative, this is still a small dataset, with 80 publicly available entries, and in a controlled setting.

Given the state of our two possible datasets we knew their size does present a strong chance of overfitting. Another complication of the small size, is the difficulty in having enough data to train and test, which can lead to overfitting as well. Since, we can not find more datasets related to our search, we do believe the best way forward is to join the two presented datasets (MMASH and APD) into a single dataset with 124 entries.

Dataset's union Both datasets contain information that is interesting to the Anxolotl Anxiety Engine. With that in mind, we will still have to take into account that the MMASH dataset contains activity, which will not be a problem in the APD, since it was recorded in a lab setting. Other than that, their duration is also different, given the MMASH has one label *per* 24 hours, while the APD contains a label *per* 2 hours.

The labels are also not interchangeable, so we must come up with a way to standardise them. To have a joint dataset, we need to have entries in common, besides the data. To that end, we are using the continuous Interbeat Interval (IBI), which can be translated to the time between two heart beats. We are also using a timeline to access every time stamp for each measurement, in both datasets. Lastly, we are saving the user's age and gender.

Regarding the labels, in the case of the MMASH dataset, the STAI result varies from 20 to 80, and can be down sampled to 3 levels, less than 30, from 31 to 49, and 50 or bigger, which can be mapped to low or no anxiety, average anxiety and extreme anxiety. Since we had a small sized dataset, we opted to downscale this label to 3 values, and so we did the same with other dataset. The APD dataset presents values from 0 to 100, with intervals of 10, without an inherent mapping, so we created ours. Our mapping intended values from 0 to 20 to be mapped to low or no anxiety, 30 to 50 to average anxiety, and 60 or more to extreme anxiety levels. This mapping was made to be close to MMASH's APD mapping, given the distances between levels.

After managing to have a single dataset with 124 entries and 3 levels of label, we start to improve the data, to allow it to achieve better results. Since the only filtering is done to the MMASH part of the dataset, no more is done after this, but now we have to calculate the features.

Dataset Filtering Even though this section is present after explaining how the datasets are joined, some data filtering is done while the datasets are separated. The activity exclusion done on the MMASH dataset is a good example of that, since it is not present on the other dataset. The rest of the filtering, such as outlier removal, is also done with the datasets separated, due to the implementation technique starting with one dataset at a time.

We present the entire flow of the dataset analysis to allow for joining, and the filtering part in Figure 4.1. Firstly we perform the filtering of entries containing certain activities, which will be explained later. Then we perform the label mapping, which was presented before. Lastly, we normalise the dataset by removing invalid values such as Nan and values of zero and below. This normalisation is only applied to the IBI entries. Lastly, we apply outlier detection to the joint dataset, which will also be explained later.

Regarding the activity detection in the MMASH, we remove three types of activity, namely, light movement, such as walking slowly, medium movement, such as fast walking or bike and heavy movement, such as going playing soccer or running. All these actives are removed because they can be detected with our wearable, and might

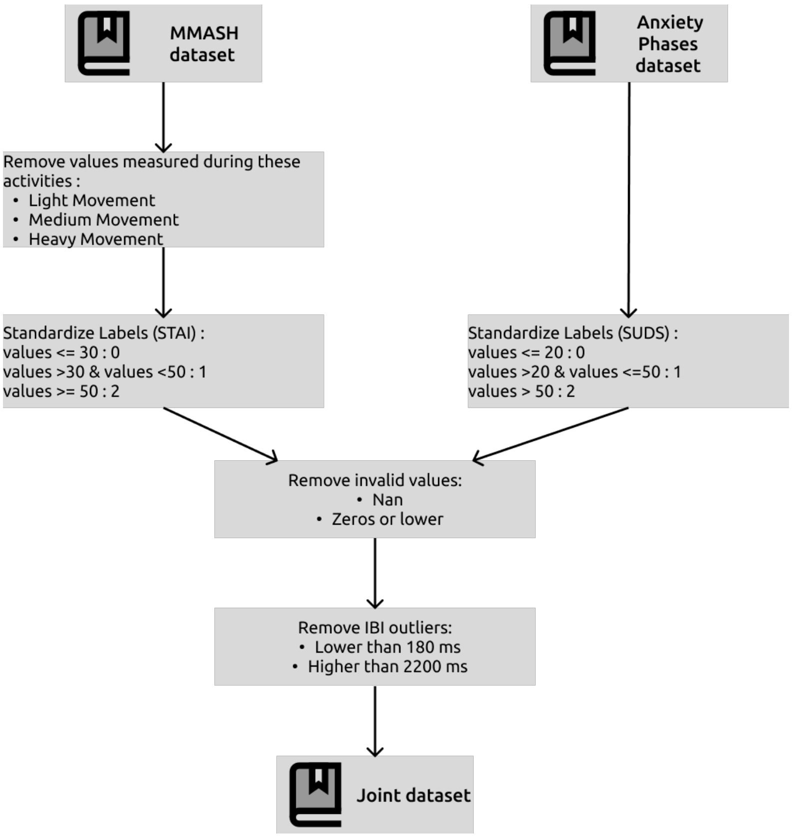

Figure 4.1: Flow to join the two datasets

skew results, both due to increased noise in the signal, due to movement, as well as a different IBI profile while exercising. Regarding the normalisation, we do not perform any particular normalisation, besides removing values of Nan, zero or negative, since the IBI can not be a negative value. Regarding the outlier detection, we picked the values of 180 ms and 2200 ms as our limits by taking inspiration from [22]. In said article, they suggest multiple ways of outlier detection for IBI, and the one we used is the fixed range of acceptable interval values; e.g., IBIs reflecting heart rates <30 (IBI - 200 ms) or >300 bpm (IBI - 2000 ms); threshold rates vary depending on the clinical population of interest [22]. Our population is one which IBI values will have a lot of impact, so to avoid mistakenly removing values, we added 10% margin of error to each limit, ending with an outlier detection between 180 ms and 2200 ms.

Lastly, we have to merge the datasets, with all the information of our interest. An IBI array with timestamps for every measurement, a label identifying the result for each measurement set and the user's gender and age. It is important to mention that we still have measurements of 24 hours (MMASH) and 2 hours (APD) in our joint dataset.

Features At this point, we have a full joint dataset, but besides the age and gender, we do not have any features regarding the IBI. Besides, we still have to fix the problem of the different durations for a single label. For that reason, our approach to the features will have to take into account all the limitations, from both the early datasets (times) and from the system itself (limited wearable sampling rate).

We have the processed raw IBI data, but we do not have access to the PPG or the ECG values. Without those, is not possible to generate some features regarding BR, but we still have a lot of information to work with. The difference in duration *per* label is the most pertaining problem, as an anxiety state from 24 hours would not be the same as one from 2 hours. Starting from the time, we apply a 1-hour sliding window, since it allows to do a good analysis of both the MMASH and APD datasets. This value also does a good job of allowing for real time detection, since we can use the data from the last hour to get a current reading. It is also important to notice, that the subjects would probably identify their anxiety levels as the highest in said day, and so the label does not pertain to the entirety of the 24 hours.

Our method for picking the better sliding window, is to divide it into two sections, the filtering and the selection. The filtering itself means that if the data for one hour does not contain at least 65% of the expected samples, that hour will be discarded from being the chosen hour to represent the label. This small filtering is implemented as a way of lowering the bias towards very sparsely populated data representing higher amplitudes, and being misinterpreted in feature creation. Upon completion, with just valid hours, we proceed to do the selection of the hour that should represent the data.

The selection starts by comparing the HRV ratio of Low Frequency (LF) over the High Frequency (HF). Pearson's correlation analysis revealed significantly positive correlations between LF/HF ratio and anxiety [54]. We also assume that the LF/HF ratio is useful to estimate the ratio between Sympathetic Nervous System (SNS) and Parasympathetic Nervous System (PNS) activity under controlled conditions. So given that anxiety is mainly controlled by the PNS, we believe that a higher ratio will correlate to higher anxiety activity, which will also be the one reported in the labels.

After this, we generate the features from the IBI from the hour that presents the highest LF/HF ratio, and we apply this selection to the entirety of the joint dataset. From that point on, we are able to achieve our features. We created a specific set of features, and the ones we are relying on have shown a significant relation to anxiety in the literature as shown in Table 4.1.

Table 4.1: Created features selection

General [52]	Interbeat Interval
Age	Mean IBI
Gender	Variability of IBI
	90th percentile of IBI
	10th percentile of IBI
	80th - 20th percentile of IBI
Interbeat Interval Time Domain [13, 58]	**Interbeat Interval Frequency Domain [34, 54, 58]**
Mean NNi	Total Power
SDNN	Very LF Power
SDSD	LF
RMSSD	HF
Median NNi	LF/HF Power
NNi 50	Normalized LF Power
pNNi 50	Normalized HF Power
NNi 20	
pNNi 20	
Total NNi range	
CVSD	
CVNNi	
Mean HR	
max HR	
min HR	
Standard Deviation of HR	
Interbeat Interval Poincaré Plot	
SD1	
SD2	
SD2/SD1 ration	

IBI Time Domain NNi: successive differences between the RR-intervals, SDNN: The standard deviation of the time interval between successive normal heart beats, SDSD: The standard deviation of differences between adjacent RR-intervals, RMSSD: The square root of the mean of the sum of the squares of differences between adjacent NN-intervals, NNi 50: Number of interval differences of successive RR-intervals greater than 50 ms, pNNi: The proportion derived by dividing NNi 50 by the total number of RR-intervals, CVSD: Coefficient of variation of successive differences, and CVNNI: Coefficient of variation equal to the ratio of SDNN divided by mean NNi.

IBI Poincaré Plot SD1 : The standard deviation of projection of the Poincaré plot on the line perpendicular to the line of identity, and SD2 : The standard deviation of the projection of the Poincaré plot on the line of identity.

All of these labels provide some value in the literature, in the case of the general features, by observational reports, and the time and frequency domain features proved themselves in experimental results. In the case of the Poincaré features, it has been used to measure stress levels with good results [31]. In our case, we try to use it inversely, meaning to try and distinguish what is stress from what is anxiety. It is also important to mention that for the generation of most features we use the HRV-analysis

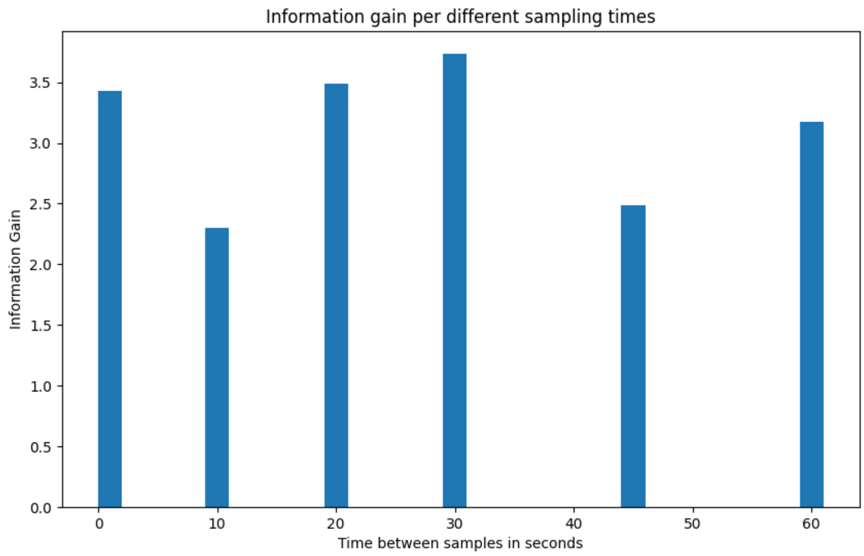

Figure 4.2: Information gain by the features on different sampling times

library [21].

After the generation of the features, we have a problem with the sampling rates, since they also are different between the datasets, 1 Hz in the MMASH dataset and 1000 Hz in the APD. Besides that, our wearable device only allows for measures up to 1 Hz, meaning we can not sample any faster than that. We also need to balance the sampling rate between accuracy and battery spent on it, given we are relying on a small battery. With that in mind, we applied a Mutual Information function, to try multiple sampling rates and see which one contains more information regarding the label.

We tried multiple sample times, from 1 to 60 seconds, because a sampling time with more than 1 minute between samples starts to harm the validity of the HRV results. As presented in the Figure 4.2, the best results were achieved by 1,20 and 30 seconds, regarding the information gain about the label using all features. We opted to go with the 30 seconds measurement for providing reliable measures, being good for battery saving, and presenting the best result.

Feature Normalisation We currently have our dataset created, albeit with values from multiple time ranges, which can hamper machine learning algorithms. We also have unbalanced classes, which can also throw havoc on the algorithms. To address this, we turned to the `SKLearn` framework, which provides multiple tools that can be used for both scenarios.

About the normalisation of the values, the group had experience with the Min-Max

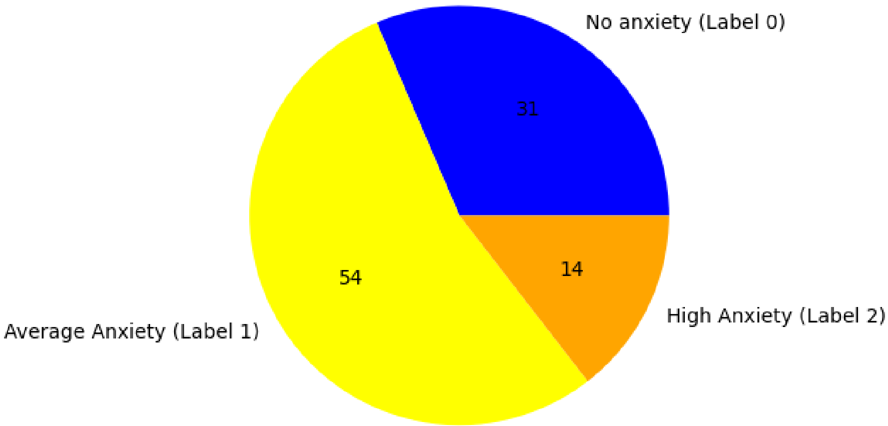

Figure 4.3: Label class distribution

scaler, which ranges all the values in the range of [0,1] or [-1,1], disregarding outliers. Given that, we are looking for the outliers in this algorithm, the Standard scaler is a better fit, since it calculates the mean of the values to 0, and scales the data according to the unit variance. Since it uses a normal curve to do it's scaling, and our values will probably fall on that curve as well, we do believe it is a better fit for our Engine.

Regarding the dataset labels, as presented before, we have 3 classes, which can be mapped to three levels of anxiety. They are divided into no anxiety (class 0), average anxiety (class 1) and high anxiety (class 2). Given the biological nature of this capture, and the fact that we are using two datasets, it would be unexpected for them to have a similar number of entries *per* class. And so, our classes are quite unbalanced. As presented in Figure 4.3, our values for all the classes are quite skewed, class 2 contains almost a fourth of the values that class 1 contains. This would lead to bad results at best, and at worst could lead to overfitting and a bad result in the real world.

To address this problem, we have two options: (1) oversampling and (2) undersampling. Our dataset has 124 entries, so undersampling is not really an option, so we have to oversample. To do that, we will use a generator that takes into account our real features, and generates new synthetic features that are similar to ours, but only with classes that we are missing (0 and 2).

Here, we are opting to use another tool provided by the `SKLearn` framework, namely Synthetic Minority Oversampling Technique (SMOTE). This tool works by selecting examples that are close in the feature space, drawing a line between them, and then fabricating a new sample in a point around that line, using the k-neighbours algorithm. It is also only used to create minority values, to balance the dataset and have a complete balanced dataset. SMOTE also has the property of being able to be applied multiple times and generate different values.

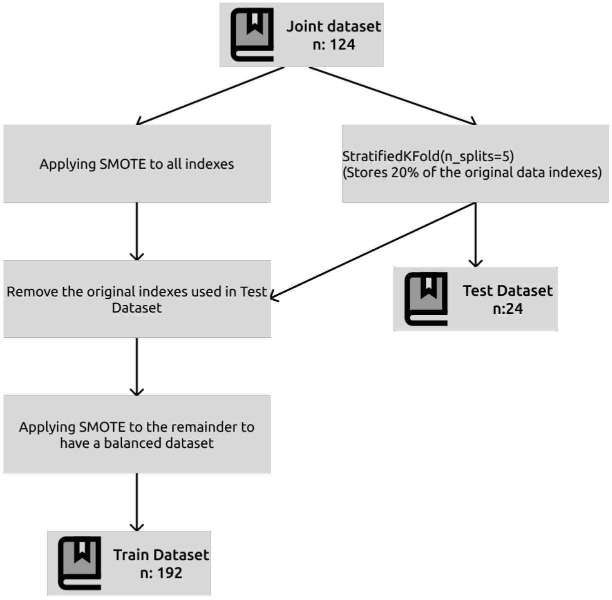

Figure 4.4: SMOTE workflow

ML-ready Data In the last section, we introduced SMOTE, but did not explain how does it fit into our algorithm. This was because we use it twice, and so its usage is intertwined with the testing and classification process. In this chapter, we will go over the training process, and the specific details of our architecture specifically targeting our small dataset as well as the tests with synthetic values. Our first approach to the SMOTE tool, was that it was good for generating new data for training, but it would be better to test with real data, since biological processes tend to be hard to emulate. But we also took the piece of advice that SMOTE can be used multiple times.

As is presented in Figure 4.4, we start by taking 20% of the original data indexes and store them for later. These values are taken with the StratifiedKFold method, to guarantee that our test dataset is representative of reality. After that we apply the SMOTE tool to all the original indexes, and finish with a perfectly balanced dataset. Since we do not want to have synthetic values on our test dataset, we simply remove the indexes, which are all from original values, from the balanced dataset, and add them up to our test dataset. Regarding the balanced SMOTE dataset, it is not balanced any longer, so we need to reapply the SMOTE tool, to have a perfectly balanced train dataset. Since this tool allows for multiple generations without needing to worry about repeated values, we do believe there is no harm in generating new synthetic values from a dataset with some synthetic values already. We also performed this feat to have the test results as a good test, meaning, our classifier has never seen them, and so will classify them as it would in the real world.

Feature selection is also nuclear to this engine, taking into account the 33 features we have, which is not a big size, but can be shortened if some pairs of features are redundant, or bring little to no value to the prediction. To this end, we are using the ANOVA, because it has a track record of usage in this subject [35], is a very lightweight algorithm which might be important in the case we send this algorithm to the mobile app in the future, and is an established method for feature extraction. ANOVA stands for analysis of variance, and checks the impact of the different features, by comparing the mean of their samples. But this is just the method, to perform the feature extraction, we then assembled a pipeline with a GridSearchCV, which runs whatever is in the pipeline with different values, and our model. Our feature testing range starts from 20 until 33, since bigger ranges seemed to take a while longer, while not providing any benefit.

4.2 Panic Attack Engine

Developing a proper engine to detect a rarely occurring *phenomena* was a bold objective of this project. While the detection of an occurrence that can take from few to several minutes is as hard as it is, the non-existence of public available datasets meant we had to get crafty. This Panic Attack Engine is presented as a very experimental endeavour, given the data viability, nevertheless it presents some potential.

4.2.1 Dataset

As stated, this development was marked by the nonexistent availability of open public datasets regarding the detection of panic attacks. While this would usually deter most researchers from pursuing development, or stop and capture a proprietary dataset, we could not afford any of those options. With that in mind, we used the theoretical knowledge amassed throughout the development of this project to propel us forward.

Panic is an extreme form of anxiety [36]. These words resonated with the panic detection cause, because we had already worked on a dataset which recorded anxiety levels in reaction to *stimuli*, and those levels were granular enough to target extreme values of anxiety. So for this effort we went back to the APD dataset.

As stated before, this dataset is quite comprehensive, and representative regarding gender, since it contains 51 females to 44 males. Age-wise is not as representative, given its focus on younger adults (18-35 years old). It was captured in a laboratory setting, and so the data might be better than what we will encounter in the real world data capturing, but that is a setback we will have to deal with.

Here we will also reiterate this dataset's form of capturing data, since in this scenario we use that information to a deeper level than in the Anxiety Engine case. The dataset itself was captured by presenting a particular stressor to each participant. The data was measured since the participants were waiting, and includes both the experiments and time outside the experiments. The experiment was divided into multiple stages, and the particularly important ones for panic detection are the instructions stage, since the participants would be aware of what was about to happen and the exposure stage, in which they would be presented with the stressor. Each dataset entry is also labelled with a timestamp regarding the starting of the experiment and the ending of the experiment.

Another important detail is that in each of the experiment stages, a SUDS questionnaire was taken, which includes the SUDS values for both the instructions stage and exposure stage. The full duration of each experiment was around 15 minutes, but the recorded data for a subject on a single experiment was around 2 hours. The data this dataset provides comes from multiple sensors, but we had to use only the data we would be relying on our developed system, which is activity context, HR and IBI, and so, in this case we are only able to use the data regarding the ECG capture.

Dataset Filtering While anxiety states tend to be more continuous and persist through time, panic states tend to be more rampant and acute. This fact influenced the way we approached the filtering of this dataset we had already used. While in the anxiety scenario we could analyse one full hour, here we are trying to have a as small as possible time frame within a reasonable duration. If the data frame is too short we can risk losing the chance to identify panic as well.

The first filtering process we apply to the Anxiety Phases Dataset in this scenario is to shorten its time. We use the timestamps provided on the dataset itself, and cut out every value that is not in between the starting and ending of the experiment. We end up with data including all the stages of the experiment, from relaxing, to the exposure, and to the small aftermath. The duration of all of this is about 15 minutes. We opted for this longer and less focused approach because in real life we will need to identify a panic state existence in an interval and not on a specific time frame that is bound to happen or not, which would be the case if we cropped the time frame between the preparation and exposure.

As presented in Figure 4.5, after cropping the timescale, our next step is to create the labels. They are created with SUDS values from the preparation stage and the exposure stage. We do this to compare them, and assume that a subject about to have a panic

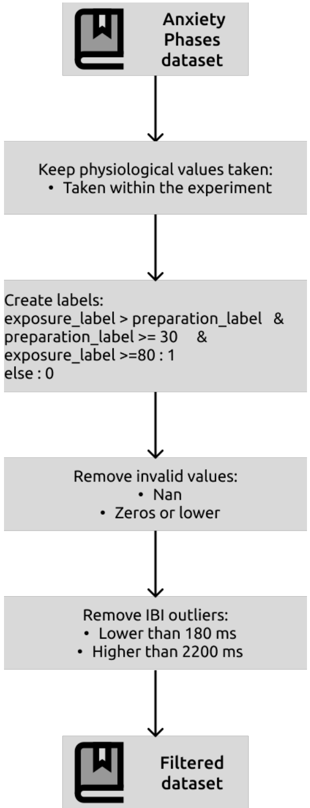

Figure 4.5: Panic Engine dataset filtering flow

attack with one of the presented stressors, would already be uncomfortable when introduced to the idea of being shown one known stressor. To evaluate this discomfort, we accept all the levels signifying average anxiety, which translates to a SUDS score of 30 or more. Regarding the happening of a panic state, we intend for it to be a heightened response when compared to the response recorded during the instructions stage, so we checked for that increase. Lastly, to truly mark the panic state as an extreme form of anxiety, we only counted as panic states, the participants who, besides fulfilling all the previous conditions, also had a SUDS score of 80 or more during the exposure. While this labelling does not guarantee that we are detecting panic, it was a close enough for our ambitions.

After these two new steps, we perform the data normalization as before, by removing Nan values and zeros, which tends to result in better ML results. We also filter the IBI data by removing the outliers of values bigger than 2200 ms and smaller than 180 ms, once again inspired by the idea that IBI is very correlated to HR and should reflect heart rates <30 (IBI - 200 ms) or >300 bpm (IBI - 2000 ms)[22]. Lastly, we do a final step of adding a 10% margin of error to avoid mistakenly removing values, ending with an outlier detection between 180 ms and 2200 ms for IBI.

Features The interpretation of panic being a severe anxiety state, made its detection that much easier, since we can now target panic in the same way we targeted anxiety, if we were able to find a way to differentiate between them.

We already had that differentiation factor, which was the LF/HF ratio, which reacts differently to the normal value in the two scenarios. With that assured, and since the features we calculated to anxiety are all related to various mental health illnesses, we will use the same features for panic detection as the presented in Table 4.1.

The Anxiety Phases Dataset presents a sampling rate of 1000 Hz, but the wearable we are using is not capable of measuring to that sampling rate, as stated before. Besides the hard limit of 1Hz for our sampling rate, we also have a preference if it would be possible to have the sampling rate of one measure per 30-sec, to keep the same frequency as our Anxiety Engine is expecting. While not mandatory, this small detail can provide an easier implementation in our system.

We once again are performing an information gain analysing using our features on different sampling times to assess if the 30-second sampling time is viable for panic detection. The results we achieved are presented in Figure 4.6. Our best results came from the 1-second sampling time, with the 30-second sampling time being a close second, close enough to be considered margin of error. The 60-second result was a surprise, since its sampling rate is quite low, considering HRV data variability per second. Our assessment hypothesis of these results is that the values can be sampled through a discrete measurement as they change their trends, and with that present a valid assessment of their continuous trends.

Given all the presented arguments, we took the chance of going again with the 30-second sampling time, in order to make this result compliant with the Anxiety Engine. Hopefully, the chance we are taking with a minimal loss in information will not be detrimental towards the development of a viable Panic Engine ML algorithm.

Feature Normalisation Real life datasets rarely come with balanced data, and this one is no exception. In this case we are classifying the data using a binary classification system which represents label 0 as No Panic and label 1 as Panic. Our distribution on the dataset is not good once again, and even though the APD uses high anxiety groups, it is not expected for all the individuals to express panic on the experiments. With that in mind, our class distribution is presented in Figure 4.6.

As we can see, our No Panic subset is much bigger than the Panic subset, almost 7-fold bigger to be exact. This raises the same problem we had in Anxiety Engine, which is little data for this Engine. In this specific case, the number of label 1 instances is not

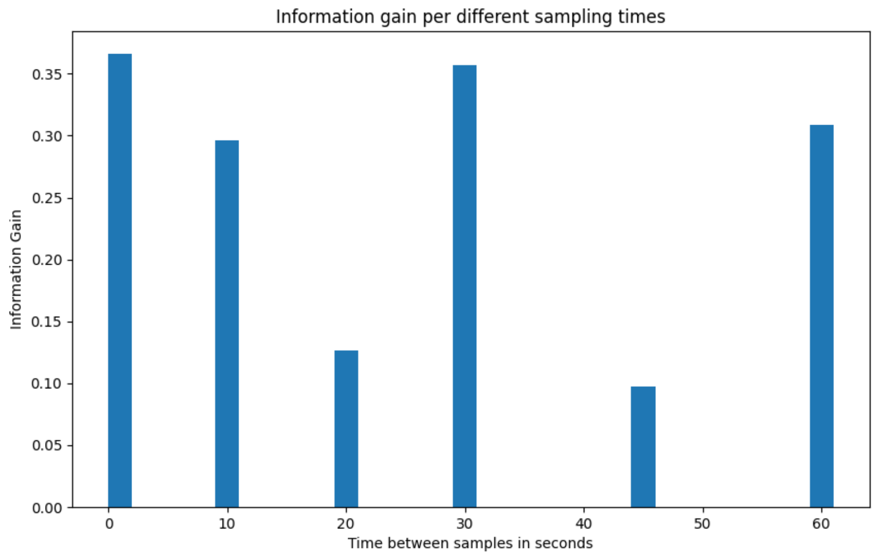

Figure 4.6: Information gain by the features on different sampling times

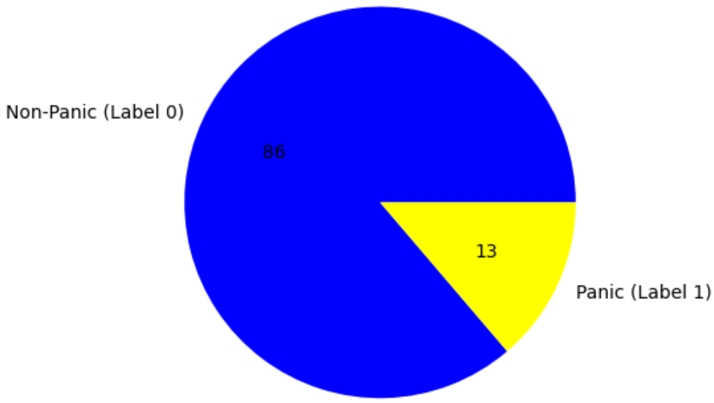

Figure 4.7: Panic label class distribution

big enough to split the original data from the training dataset and then test on purely original data, as shown in Figure 4.7. That approach would either lead to a case where our Test dataset had little label 1 instances, or another one where our Train dataset would have no original data to train, and our model would be trained on just synthetic data. Both of these cases would not be good to our future results.

For that reason, we once again are turning to oversampling techniques, and once again we are using the `SKLearn` provided feature named SMOTE. Since it provided good results in our other endeavour, we went ahead and tried to balance our dataset using it. Our dataset contains 69 instances of label 0 and 11 instances of label 1, adding up to 80 total instances. With the help of SMOTE we are able to achieve a dataset of 138 instances, which is not a big dataset, but it certainly will convey better values witha

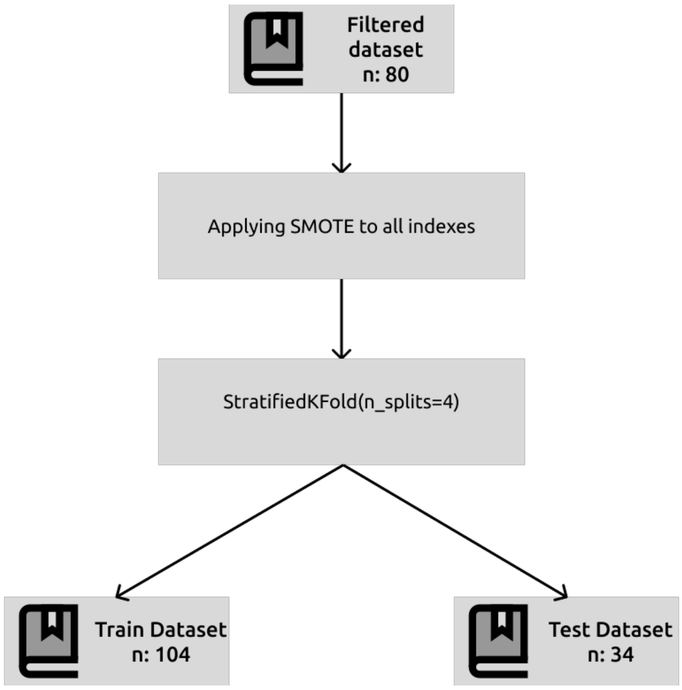

Figure 4.8: Panic Train and Test dataset creation

bigger dataset.

ML-ready Data The presence of a single dataset eases our processing needs in this case. While our data quantity is smaller, it is also easier to apply and less complex to work with. This is reflected throughout the Panic Engine. Instead of performing multiple detours and ways to try and fix the data in a cohesive package, APD allowed us to simply breeze through it.

Figure 4.8 presents us with the flow we are using to reach our Train and Test datasets from the starting Filtered Dataset. As stated before, we apply `SKLearn`'s SMOTE algorithm to oversample our data in a synthetic manner, which balances our dataset. Then we split that balanced dataset into 4 splits, and use 3 of them for the Train dataset and one for the Test dataset. We are able to use 25% of our data to test the algorithm while the rest is used to train it. While it is a small amount of data, once again we are able to test our model with data unknown to it, which is expected to convey more realistic values. It is important to remember that this scenario is more inclined to overfitting since the Test dataset is entirely new to the model, but most of its label 1 instances are synthetic, and that might skew the results.

As before, we also end up with 33 features, which depending on the cases might be interpreted as a big amount. We once again, are using the ANOVA method to perform feature selection, due to its good record in the HRV feature selection [35]. Our way of performing the test and classification does not change in this case, as we use a pipeline supported by GridSearchCV and contains the ANOVA feature selector, which selects from 20 to all the features (33), and the model that we are intending to train and test, which can be changed.

5

Machine Learning and Engine results

Usually the results are the most significant aspect of a project, but in this case we would tend to disagree. While our results are quite good, they are not able to reveal all the work done on building the system. They are imperative for its viability, yet they alone are incapable of providing the intended value to people's lives.

This chapter will be divided into the main components results, and will have a broader case regarding the Engines results, which are more complex, and more suitable to presented as results. Contrastingly, the system results will be more theoretical regarding the achieved abilities of our system, with very few practical results, most of which will come from usability scenarios.

After the presentation of our results, we will present an analysis on our results, and what our work was able to achieve in the form of a discussion. This discussion will touch points such as the viability of the results, as well as parts of the implementation that could have been better. Limitations found in our project will also be presented.

5.1 Machine Learning Results

5.1.1 Anxiety Engine

In this topic, the features are used to try and distinguish between three levels of anxiety in a person. While we have some limitations regarding the data set, we designed a ML

Table 5.1: Anxiety Engine results from different classifiers, using the train and test data set.

Methods	Train set Accuracy	Test set Accuracy	Test set F1-score
SVC(rbf)	**0.85**	**0.92**	**0.90**
Random Forest	0.78	0.83	0.82
XGBClassifier	0.77	0.83	0.82
Decision Tree	0.71	0.79	0.79
K-neighbors	0.77	0.75	0.74
Logistic Regr.	0.69	0.70	0.71

flow around the data that we have, specifically targeted for this purpose. It receives the data and outputs a list of labels classifying anxiety with a multi-class classification. The pseudocode is presented in Algorithm 1.

Algorithm 1 Workflow for the supervised classification

```
jointDataset= joinDatasets("MMASH","APD")
normalizeDataset(jointDataset)
trainDataset, testDataset = smoteWorkflow(jointDataset)
clf = Classifier
selector = ANOVA

pipe = Pipe(selector, clf, scoring="acc")
for fIndex in range(20,33)
pipe.fit(trainDataset[fIndex])
pipe.predict(trainDatasetLabel[fIndex])

pipe.fit(trainDataset[selector.bestParams])
pipe.predict(testDataset)
```

As explained before, we start by joining the two datasets, MMASH and APD, and apply some operations to guarantee they work together. These operations are the label normalisation and feature removal for both of them. After that, we normalise the dataset by removing invalid values and normalise their values with the `StandardScaler`.

Entering the train and testing stages, we apply the `smoteWorkflow`, as presented in Figure 4.4 and end up with both the train and test datasets. Now, we have the classifier decision stage, in which we will try multiple classifiers, most of them referenced in the State of the Art analysis, and some new.

Our testing is done with a k-fold selector, with 5 splits and 5 repetitions, to ensure a good replicability in our experiments, while not presenting a one off good value. Our findings are presented in the Table 5.1.

From these results, the most surprising was the Decision Tree, which being a very simple classifier, managed to be close to the ensemble classifiers in terms of test accuracy. The XGBClassifier was added, since currently is a trending classifier, and we tried to see if it performed well in our case, and it delivered quite good results. These results could probably have been better, but we were not able to tune this specific classifier properly, since it is not supported by the `SKLearn` framework.

Regarding the tuning, we were able to use the aforementioned pipeline to tune our classifiers. We did not hyper tune them, but we tuned the values of 3 or 4 arguments for all of them. In the case of our best result, the Support Vector Classifier (SVC), the tuning parameters used were max iterations to 1 000 000, to avoid problems of not fitting during the standard iterations, C=80, tol=0.5, gamma="auto", kernel="rbf" and features set to 32. All of these values were tuned automatically, except the max iterations parameter.

Our best result was the SVC with the "rbf" kernel. All along the development was consistently outperforming the rest. This classifier has some usage in biological data classifying as *per* our findings, and it was also mentioned before as one of the best in the State of the Art review. Our result of 92% accuracy is particularly impressive, given the small size of the dataset, and the usage of SMOTE. But all throughout the testing, the values from this classifier and the others seemed to hold up, and so, we do believe that even if this result might be a bit skewed due to a small test dataset, a real world usage would not be far from it.

5.1.2 Panic Engine

The panic engine was one of the latter parts of work to be performed, and so it relied more on a heavily theoretical foundation, rather than trying and figuring out what worked and what did not. This is transpired throughout its development, which presented few hiccups, while providing a quite boring, although straightforward implementation.

This approach is visible in the pseudocode presented in Algorithm 2, which is strangely similar to the Anxiety Engine pseudocode (see Algorithm 1), given they are measuring slightly different mental health metrics. The main differences between them are, in the processing of the data, relative to the time frame allowed per label and the label creation, which are two totally different processes. In the case of the panic engine, only

Algorithm 2 Workflow for the supervised classification

```
apdDataset= joinDatasets("APD")
normalizeDataset(apdDataset)
trainDataset, testDataset = createTrainTestDatasets(apdDataset)
clf = Classifier
selector = ANOVA

pipe = Pipe(selector, clf, scoring="acc")
for fIndex in range(20,33)
pipe.fit(trainDataset[fIndex])
pipe.predict(trainDatasetLabel[fIndex])

pipe.fit(trainDataset[selector.bestParams])
pipe.predict(testDataset)
```

Table 5.2: Panic Engine results from different classifiers, using the train and test data set.

Methods	Train set Accuracy	Test set Accuracy	Test set F1-score
SVC(rbf)	**0.94**	**0.94**	**0.94**
Random Forest	0.90	**0.94**	**0.94**
XGBClassifier	0.88	0.91	0.91
Logistic Regr.	0.87	0.85	0.85
Decision Tree	0.87	0.76	0.76
K-neighbors	0.82	0.71	0.69

one dataset is used, which is the Anxiety Phases Dataset, and this eased its whole ML implementation.

After performing the normalisation and creating the Train and Test datasets, we use the pipe with the ANOVA method feature selector to test for the best feature selection. The pipe once again allows adding multiple arguments to tune the different ML algorithms. Our results with a few tuned arguments are presented in Table 5.2.

We are expecting the results to be slightly better in this case, given that we are using a binary classification in this case, trying to distinguish between Panic and Non-Panic states. It is also expected to see good results from the algorithms we have used in the anxiety engine, since, as stated, panic is an extreme form of anxiety, and so their good results should translate to this engine as well.

The results presented for this engine were surprisingly good, and while we could expect some improvement due to the binary nature of the labels, worsening results were also a possibility, given the liberties taken on labelling the Panic States. We were specially surprised by getting results that consistently surpassed the accuracy levels presented in the Anxiety engine. Once again, we tuned all the classifiers to some extent using automatic tuning, except for the XGBClassifier.

It is a very valid possibility that if the `SKLearn` library provided a feature to automatically tune the XGBClassifier, we would be able to improve our accuracy, but since our results were good for our intent, we did not pursue that effort. Regarding our best result, the SVC, we once again set a higher number of max iterations to avoid problems with non converging values. In the automatic feature tuning, the best ones parameters for this classifier were C=75, tol=0.05, gamma="auto" and kernel="rbf", which are similar features to the ones presented in the anxiety engine. Regarding the features, this model uses all of our 33 features.

We can conclude that this result of 94% using SVC with a "rbf" kernel worked well for both scenarios, because panic can be seen as an extension of anxiety levels. The fact that results are not exactly equal increases our trust that we are targeting slightly different things, although not unrelated. These results give us some trust that this classifier can be used in the Anxolotl System to detect Panic States, but probably not PAs.

5.1.3 System of the Anxolotl App

The question to answer here would be *What can this system achieve?*, while we can not answer that without a full deploy, and some real life testing, we can infer that from the work done. These results will be mostly theoretical on the many ways this system can live up to its expectations. There will also be some real life results from the battery and the usability aspect of this app.

The under the hood aspect of our system was working as expected in the tests performed during development, no major hiccups during the usability. Overall, the experience provided by the Anxolotl App could have been mistaken by a professional application. The two supported languages (English and Portuguese) are fully integrated into the app's work flow. In its current state, the app could be deployed in the real world, if the amount of users was low, since upon a higher load, the system would probably crumble due to lack of resources. After using the Anxolotl App for a full day, we can reliably affirm that it can work for a full day in a headless mode, just relying on the notification. Upon opening the application, we are greeted with our current metrics, and are able to use the Wellbeing exercises without interrupting the working cycle of the physiological data capturing.

Table 5.3: Wearable battery over time during usage

Time [hour]	Battery consumption [%]
00:00	100%
12:00	85%
24:00	72%
36:00	60%
48:00	42%
60:00	31%
72:00	17%
80:20	0%

Regarding the impact of an app always running in the background, the test phone (Google Pixel 4a 5G) presented a 3% consumption during a 24-hour period. The phone presented a battery capacity of 3885 mAh, which comes around a spending of 116 mAh during 24 hours. For a full day, the app would spend around 174 mAh, which is a very workable amount of battery, since most phones nowadays contain a battery at least 20-fold that amount. Most users would probably not notice our app running in the background. The wearable end on the other side, is also an important factor. The battery spent by it might make or break the viability of this whole project. The group performed a test, letting the app run uninterrupted until the wearable battery ran out, and our results are presented in Table 5.3.

Regarding the usage of the wearable, some pauses were taken from wearing the wearable during testing, such as when having a bath, cleaning the dishes, and one-off situations. It is expected that the wearable kept working during those times, as the Anxolotl App does not stop the measuring when the wearable is detected as not being worn. The wearable was taking a measurement every 30 seconds as its sampling rate.

From these results, we can say that the battery lasts comfortably up to 3 days with our intended usage. This presents a promising result, since that value would make using the wearable more attractive, since it is not another daily charge gadget. While this result is not the greatest, which we would rate that as a 7-day battery duration, it still is quite good and usable on a daily scenario, and does not seem to present the Achilles heel of this project, which could be the case if the duration was around 1 day, which would decrease the appeal of this system.

5.2 Discussion

This study presents a fully integrated system to classify multilevel trait anxiety and detect the existence of panic states on a daily basis in the real world, and to the best of our knowledge is the first system that sought to do that. While research into anxiety measurement has been conducted, it mostly happens on a laboratorial setting, or if not, it is done in extremely specifics scenarios.

Regarding the achievement of the presented results, we would like to add, that all of the development was done in two machines, a laptop running Manjaro OS, with a Linux Kernel version 5.15 and a Windows 10 desktop. The laptop was used mainly to develop the software and algorithms, while the Windows 10 desktop was used to generate the features, while the testing was conducted on the Linux laptop. This is an important aspect, since the generation of datasets in the Linux machine yields different results from the ones presented in this paper.

With our engines made and tested, we were missing the implementation of a full system that could detect and inform users in real time, about their anxiety levels, and panic state. In this section we will discuss the multiple parts of our work, starting from the engines until the system itself. We will go over some similar papers to parts of this work, as well as talking about limitations in the choices made.

5.2.1 Engines

While our anxiety engine was one of the flasghips of our system, it was not the only important aspect, and was not treated as such. With that in mind, is with great joy that we can affirm that our algorithm seems to present some of the best results in this area of multi-level anxiety classification.

According to our research, the closest system, even though hypothetical, would be [38], which suggested an algorithm to detect anxiety on autism spectrum disorder patients. Their main method was to develop a Kalman-like filter, to integrate into HR and ACC signals, which would be able to just take anxiety levels whenever the activity data would not trigger false positives. Their detection accuracy was of 93%, but with a binary anxiety indicator.

On a multi-class classification of anxiety, we have the study by Zheng et al. [57], which using a wearable detecting EEG and PPG, managed to achieve a 62% accuracy. On the other side of the multi class classification, we found a more recent study, in which

Arsalant et al. [1], managed to achieve a impressive 83% anxiety detection score for three class anxiety classification using an ECG sensor.

Even though our results are quite promising, they must be evaluated with a critical eye. The usage of two datasets, may not make the labelling good enough for real world usage. With a single dataset we could have more trust in our results. It should also be noted that the mapping of the two dataset's labels into a three level anxiety was based on theoretical principles of people without a theoretical degree in the area, and so the results must be taken with a grain of salt. To present this results in a full trustworthy manner, we would have to do design and perform a clinical trial.

Another important point about the results was the small size of the test dataset, which might have skewed the results, even though, the train dataset stands up to scrutiny and all of our classifiers presented somewhat close results. These nearby values bring more confidence to our work, since it does not seem to be a fluke, yet these limitations are to be taken seriously.

The usage of synthetic data was a great idea, since it allowed our anxiety engine to perform on a bigger amount of data, improving our results. On the downside, is another factor that brings uncertainty to our results, since synthetic data is not real data, and with that comes said uncertainty.

On a bright note, our intended usage of context, namely in the removal of activity that we considered harmful to our goal, was a great idea, and helped to make the results better. Another great point was the IBI filtering, which seemed to be good enough to filter out most noise while keeping the real data on the dataset. This was a bit unexpected specially with the application in both datasets, even on the APD, which was captured on a laboratory setting, but they both did great result wise. The labelling by hour with the biggest LF/HF ratio also seemed like a winning bet. Its correlation with anxiety has some research backing it, but it is not a fact, and that theoretical grounding was one of our conducting lines in the filtering and the feature creation as well.

The Panic Engine was challenging to build, since we did not have access to open publicly available datasets that targeted the occurrence of PAs on people. That on itself was a major hurdle that the group had to get through, and it was solved by taking a workaround. We used the basic idea of Panic being an extreme form of anxiety [36]. And from there extreme anxiety was extrapolated from the APD dataset when a stressor was presented. While this does not guarantee a PA happened, we interpreted these results as the proof that the participant was at least in a state of panic.

Given this reasoning, our Panic Engine is likely better at targeting panic states and not PAs, which is not exactly what was proposed in the objectives. With that in mind, we do believe, the results were positive, given the data circumstances. We are able to identify panic states with a 94% accuracy score, which is a very promising result, suitable for a real world trial.

Still analysing the results, the good and the best ones are all quite close, with a small difference in the training dataset in the case of the best two ML algorithms. We interpret this as a case of a good choice of data, and good filtering result, and while our data was not near perfect regarding our intended label, we definitely isolated data that can be well classified. We present these results with the confidence that our Panic Engine is capable of detecting Panic states, while leaving the question open for the case of acute Panic Attacks.

Regarding the two Engines, they both used the Anxiety Phases Dataset, but since the filtering, selection and overall datasets were different, we do not believe that this will pose a problem. We do have a big reliance on this dataset, which might be problematic, if for example, it contains any systematic error. We do believe that, for this case, a proof of concept system, this does not present a viability problem, since it was the only feasible way to prove the system's concept, but it is a concern that should be addressed in the near future.

Lastly, it was also insightful to use a lightweight feature selector in both cases, since, if this algorithm is shipped directly on the app, it will make the usability much faster. With that, would also come better battery usage on the mobile app front, which is a great secondary effect.

5.2.2 System: Anxolotl app

Regarding our system, it was made and implemented as a proof of concept system, but ended up more closely resembling a Beta version of a product to be deployed. While our quality standard would not stand up to it, given the lack of unit testing, and repeatability in said testing, the quality of the final product is above average for what a lone developer should be able to achieve. With some tweaks, the system could be deployed, although it would miss several key maintenance components such as loggers and watchdogs.

Starting with the user facing component, the Anxolotl App. It could have supported multiple wearable devices, if the implementation of generic Bluetooth Consortium services was more widespread. Due to the lack of usage in that area, we did not implement a generic wearable interface, which would have been interesting to allow the usage of our solution by the public. With the work we have now, adding support for generic bands would be easy, but we could only support the generic HR service, which would hinder our app's viability, given its reliance on physical activity context.

The lack of customisation on the wearable was also a problem, while it provided an API that we used, it presented some errors and inconsistencies. Even their documentation did not add up with the real life, and we had to develop a workaround for the discrete measurements. On the bright side, our workaround saved battery, since we set the band on standby in between measurements, but the lack of a fully supported solution can bring instability further down the road.

Regarding the mobile app, its features allow for a seamless experience, from the dashboard to interact with the values, to the wellbeing exercises, up until the profile section, our app simply provides an experience that works. Its design and user experience was something that was purposefully built by taking hints from apps with similar focuses such as Google Fit[1], Headspace[2] and Zepp Life[3]. We intended to create a mobile application that people could and would want to use, and we think we achieved that objective.

Our app also allows for multiple users logged in at the same time, even though that is not a feature we would be comfortable advertising, since our app is intended for one person *per* smartphone. We had to implement this, since it is a possibility in an authentication system with multiple authentication providers. Moreover, the target is both Android and iOS, but due to technical limitations, we were unable to test it on iOS, since it implies having both a MacBook and an iPhone, which the main developer did not have. For that reason it is not possible to assure that our app will run on iOS, although every development step of the app was made having in mind the compatibility, and as such, no plugin was used that was incompatible with iOS.

One impactful event was the development of the notification in the app, due to its inherent complexity of thread changing, and sharing resources, all while still working in a headless mode. While little, this was a nuclear feature, took too long, and was quite hard to develop, since it was not intended to be used as it was designed to be used, given the mobile operating systems tend to terminate applications that run nonstop on the background, since they have a tendency to spend too much battery. We could avoid both of these pitfalls, but in the end, developing the notification flow was still quite the challenge.

Our message broker was an important piece to achieve some of our so-called values.

[1] `https://www.google.com/fit/`. Accessed on September, 2022.
[2] `https://www.headspace.com/`. Accessed on September, 2022.
[3] `https://www.zepp.com/`. Accessed on September, 2022.

With it, the group was able to achieve decoupling, the possibility of scalability and security through TLS. While, for the system that is presented here, a simple HTTPS connection between the Anxolotl App and the datacenter would serve the purpose, we did believe the future proofing was worth it. With the addition of the broker, we were also able to avoid some concurrency problems, particularly in having the server busy doing a computationally intensive task, while an instance of the app was trying to upload values.

On one hand, our implementation of the message broker might even seem over-engineered to a proof of concept system. On the other hand, the need to present a viable system, which also means a resilient and reliable system pushed our choice to a better alternative in the face of slow calculations. With this solution, we are able to scale the number of datacenters if needed, while providing a trustworthy messaging channel between the app and the datacenter.

Lastly on our system, the datacenter, was devised to calculate values and store physiological data. The datacenter is composed by two parts, the server that is stateless and the database, which contains the data to be analysed. Since we were able to create a stateless backend, we went the extra way to provide one thread for each queue listener. This was developed as a speed optimization, but even with this adaptation, it might end up becoming slow with 30 or more users. In such case, we can either increase the number of threads or increase the number of datacenter server instances of each analysis type (panic or anxiety).

While the number of instances should be increased, the database must be kept at one instance per analysis type in an initial horizontal scaling scenario, to avoid the need for a load balancer. In the case of the implementation of a load balancer, that could distinguish which users were processed by each datacenter, we could scale each datacenter as much as needed. The existence of a single database for multiple instances is not a problem of itself, given the accesses to the database are controlled in the current state, to avoid concurrency errors, and so accesses from multiple instances would not behave differently from accesses from multiple threads.

In the case of future datacenter instance scaling, the backend instances will have to be decoupled from the database, meaning one database instance will have multiple backends. This is a conscious choice to avoid the implementation of a load balancer, given that it would increase system's complexity. The presented implementation has the access to the database controlled in the perspective of concurrency, and so we could increase the number of backend instances without raising concurrency problems. In other words, the current accesses we have from multiple threads do not behave differently from multiple accesses coming from multiple instances.

The database itself could have been more robust, and perhaps implementing a distributed architecture providing multiple distributed instances. That was where the group drew the line. The complexity involved would not justify such an endeavour in this proof-of-concept work. In general, we do not have strict rules about storing the data since the primary point is to store it for future analysis. Additionally, we believe that storing the filtered data helped us achieve our data objectives, perhaps even more than anticipated.

Lastly, the whole system implementation was better than expected quality wise, on the other side, it felt short on overall characteristics, since no proposed additional milestones were achieved. On that regard, we do believe our goals were too ambitious at the beginning of this project, and the under researched mental health level detection, increased the level of difficulty in the ML development more than the group expected.

6 Conclusions

In this last chapter, we will go over what is the work that could be build upon this project, since it still could use further development. While we do believe our work is almost ready to be shipped, we also think it could use more flare to increase its appeal. Lastly, we will close this thesis with a final and succinct evaluation of the developed project, while trying to present a view that includes both some positive and negative outlooks in an unbiased manner.

6.1 Conclusion

Our system was set out to be safe, scalable, modular and present some fault tolerance. Analysing each of these attributes, we do believe we achieved it. We can guarantee an end-to-end encryption, with the usage of TLS, the safety of the stored physiological data with the App Secure Storage and lastly the safe storage of said data in an isolated datacenter. Regarding the scalability, our RabbitMQ broker is able to support multiple smartphone users at the same time, as well as multiple datacenters performing work, so scalability was implemented to a degree as well, although it lacked proper testing. On the fault tolerance side, wehave few reasons to fear losing the physiological data, since all the systems have implemented ways around the next component failing, be it local storage on the App or maintaining the data in queue in the broker case. Lastly, the modularity, while we coupled the database with the server, all of our components are modular, as we can switch every part of the system without affecting the whole flow, as we have done before when switching the Mosquitto implementation by our current RabbitMQ implementation, and our system did tolerate well that change. Of course, the ways of connecting are different for different implementations, but the flow does not need to suffer changes, providing some feature proofing.

Stability wise, according to our tests, our system as been able to perform as a unit, which is an achievement, given the pioneer nature of this system. We tried running a testing scenario with one instance of each system component, and we achieved good results. Our system was devised with the intention of running multiple datacenters with different ML algorithms, and while that is not happening currently, nor tested, it was devised to accomodate for that.

Regarding the Machine Learning development, on the Anxiety Engine, we were able to develop a system that could distinguish between 3 Anxiety levels with a result of 92% accuracy and a F1-Score of 90% which is particularly impressive, given the small size of the dataset, and the usage of synthetic data. On the Panic Engine, we presented a system that could reliably differentiate between the states of Panic and Non-Panic with an accuracy of 94% and a F1-Score of also 94%. We are taking this result with a grain of salt, due to a small dataset and various other technical limitations, but the group believes this result can hold up in the real world.

The possibility of helping an under researched group with this tool, made all the development worthwhile. While the awareness around mental health issues has been increasing, PD, GAD, and stress have not seen an increase in resources. The development of a continuous, accessible and non-invasive anxiety level monitor is a challenging yet promising field. The trend in the wearable scenario, is for them to include more and more sensors, and with that, new opportunities to track mental health issues. With that more research opportunities will arise, hopefully making the application of this aspect of research ever more accessible to public.

6.2 Future Work

Regarding the future work, most of it would be related to the fine tune of the user experience in the way of features, or at least it would end up as such. Most of the presented suggestions would be quickly achievable, but due to many unexpected hiccups throughout the development of the system, and a bigger effort in providing quality solutions and features that can endure long times working without fatally failing, they were put on the backseat.

Implementing the code for a generic wearable would be a great step to increasing our potential user base. While its implementation with just HR data would be easy, the implementation with the activity service would present a more formidable challenge, since we do not have the hardware for that endeavour. In either case, it would be worthwhile, given that we could help a bigger part of the population.

On the application front, we would like to add more information to the dashboard other than a monthly anxiety level information, current anxiety level information and daily highest anxiety level information, and panic statistics. This could be increased by adding a daily hourly view, as well as allowing the monthly scale to see the history since starting the usage of the app. Still regarding the app, testing on iOS would be an immediate future work as the app was developed with that limitation, but it was not possible due to technical limitations as stated. Lastly, a feature that could be added to severely improve the quality of life of the app, would be notifications for the events generated, since that would better grab our user's attention to anything we deemed as important.

Given our background and participation on a Stress Detection Challenge, we would also like to add stress detection to our *repertoire* of ML algorithms. This was not possible due to a lack of time, but presenting stress levels, anxiety levels and panic attacks in a single application would present the Anxolotl App as the Swiss knife of mental health monitoring. Another suggested, but not achieved optional objective that we would like to see implemented in the future is the prevision of panic attacks with up to one hour of anticipation. We were not able to achieve this, due to lack of datasets in this regard, but this feature would significantly positively impact the lives of people that suffer from PD. Regarding the datasets, we do believe that a next step would be to capture our own datasets concerning anxiety and panic attack detection. Given the small size of the ones used in this work, capturing those datasets would bring more trust to our presented ML algorithms, if we could test them effectively on new datasets. If everything went according to plan, a clinical trial would also be interesting from the view point of having medical professionals assessing the viability of this tool on a value confidence scale.

Lastly, an optimisation feature that could be implemented in system would be to transport the machine learning algorithms to the app. While this would be a very future oriented goal, which would severely decrease the choke points of our system. We would still like to capture the physiological data, to keep improving the models, but in that scenario the lack of network by the user would not mean that the app stopping the update of new mental health metrics.

References

[1] Aamir Arsalan and Muhammad Majid, "A study on multi-class anxiety detection using wearable eeg headband", *Journal of Ambient Intelligence and Humanized Computing*, pages 1–11, 2021.

[2] Borwin Bandelow, Sophie Michaelis, and Dirk Wedekind, "Treatment of anxiety disorders", *Dialogues in clinical neuroscience*, vol. 19, no. 2, page 93, 2017.

[3] Leonard Bickman, "Improving mental health services: A 50-year journey from randomized experiments to artificial intelligence and precision mental health", *Administration and Policy in Mental Health and Mental Health Services Research*, vol. 47, Sep. 2020. DOI: 10.1007/s10488-020-01065-8.

[4] Daniela Caldirola and Giampaolo Perna, "Toward a personalized therapy for panic disorder: Preliminary considerations from a work in progress", *Neuropsychiatric Disease and Treatment*, vol. Volume 15, pages 1957–1970, Jul. 2019. DOI: 10.2147/NDT.S174433.

[5] Adrienne Chung, Richard Gevirtz, Raouf Gharbo, Melinda Thiam, and J.P. Ginsberg, "Pilot study on reducing symptoms of anxiety with a heart rate variability biofeedback wearable and remote stress management coach", *Applied Psychophysiology and Biofeedback*, vol. 46, Dec. 2021. DOI: 10.1007/s10484-021-09519-x.

[6] Michelle Craske, Scott Rauch, Robert Ursano, Jason Prenoveau, Daniel Pine, and Richard Zinbarg, "What is an anxiety disorder?", *Depression and anxiety*, vol. 26, pages 1066–85, Dec. 2009. DOI: 10.1002/da.20633.

[7] Luís Cruz, Jonathan Rubin, Rui Abreu, Shane Ahern, Hoda Eldardiry, and Daniel Bobrow, "A wearable and mobile intervention delivery system for individuals

with panic disorder", Nov. 2015, pages 175–182. DOI: 10.1145/2836041.2836058.

[8] Lara Ebenfeld, Stephan Kleine-Stegemann, Dirk Lehr, David Ebert, Burkhardt Funk, and Matthias Berking, "A mobile application for panic disorder and agoraphobia: Insights from a multi-methods feasibility study", *Internet Interventions*, vol. 19, page 100 296, Nov. 2019. DOI: 10.1016/j.invent.2019.100296.

[9] Lara Ebenfeld, Dirk Lehr, David Daniel Ebert, Stefan Kleine Stegemann, Heleen Riper, Burkhardt Funk, and Matthias Berking, "Evaluating a hybrid web-based training program for panic disorder and agoraphobia: Randomized controlled trial", *Journal of Medical Internet Research*, vol. 23, 2021.

[10] Mohamed Elgendi and Carlo Menon, "Assessing anxiety disorders using wearable devices: Challenges and future directions", *Brain Sciences*, vol. 9, page 50, Mar. 2019. DOI: 10.3390/brainsci9030050.

[11] Peter Morgan. [Online]. Available: https://www.freemindfulness.org/.

[12] Christopher Gale and Mark Oakley Browne, "Generalized anxiety disorder", *Am Fam Physician*, vol. 67, pages 135–8, Feb. 2003.

[13] Shruti Gedam and Sanchita Paul, "A review on mental stress detection using wearable sensors and machine learning techniques", *IEEE Access*, vol. 9, pages 84 045–84 066, 2021. DOI: 10.1109/ACCESS.2021.3085502.

[14] Alan J Gelenberg, "Psychiatric and somatic markers of anxiety: Identification and pharmacologic treatment", *Primary care companion to the Journal of clinical psychiatry*, vol. 2, no. 2, page 49, 2000.

[15] Martin Gjoreski, Mitja Luštrek, Matjaž Gams, and Hristijan Gjoreski, "Monitoring stress with a wrist device using context", *Journal of Biomedical Informatics*, vol. 73, pages 159–170, 2017, ISSN: 1532-0464. DOI: https://doi.org/10.1016/j.jbi.2017.08.006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1532046417301855.

[16] Carlos Mendes, Ricardo Correia, Guido Santos, Rafael Cecílio, Nuno Gomes, and Matilde Pato, "A platform for medical remote monitoring of patients with chronic diseases", Sep. 2021.

[17] Hee Jeong Han, Sina Labbaf, Jessica Borelli, Nikil Dutt, and Amir M. Rahmani, "Objective stress monitoring based on wearable sensors in everyday settings", *Journal of Medical Engineering & Technology*, vol. 44, Apr. 2020. DOI: 10.1080/03091902.2020.1759707.

[18] Jennifer Healey and Rosalind Picard, "Detecting stress during real-world driving tasks using physiological sensors", *Intelligent Transportation Systems, IEEE Transactions on*, vol. 6, pages 156 –166, Jul. 2005. DOI: 10.1109/TITS.2005.848368.

[19] Judith Held, Andreea Vîslă, Christine Wolfer, Nadine Messerli-Bürgy, and Christoph Flückiger, "Heart rate variability change during a stressful cognitive task in individuals with anxiety and control participants", *BMC Psychology*, vol. 9, no. 1, page 44, 2021, ISSN: 2050-7283. DOI: 10.1186/s40359-021-00551-4. [Online]. Available: https://doi.org/10.1186/s40359-021-00551-4.

[20] Blake Anthony Hickey, Taryn Chalmers, Phillip Newton, Chin-Teng Lin, David Sibbritt, Craig S. McLachlan, Roderick Clifton-Bligh, John Morley, and Sara Lal, "Smart devices and wearable technologies to detect and monitor mental health conditions and stress: A systematic review", *Sensors*, vol. 21, no. 10, page 3461, 2021, ISSN: 1424-8220. DOI: 10.3390/s21103461. [Online]. Available: http://dx.doi.org/10.3390/s21103461.

[21] Robin Champseix. [Online]. Available: https://github.com/Aura-healthcare/hrv-analysis.

[22] Kathi J. Kemper, Craig Hamilton, and Mike Atkinson, "Heart rate variability: Impact of differences in outlier identification and management strategies on common measures in three clinical populations", *Pediatric Research*, vol. 62, no. 3, pages 337–342, 2007, ISSN: 1530-0447. DOI: 10.1203/PDR.0b013e318123fbcc. [Online]. Available: https://doi.org/10.1203/PDR.0b013e318123fbcc.

[23] D Jolene Kinley, John R Walker, Murray W Enns, and Jitender Sareen, "Panic attacks as a risk for later psychopathology: Results from a nationally representative survey", *Depression and anxiety*, vol. 28, no. 5, pages 412–419, 2011.

[24] Antonia Kotianova, Michal Kotian, Milos Slepecky, Michaela Chupacova, Jan Prasko, and Ingrid Tonhajzerova, "The differences between patients with panic disorder and healthy controls in psychophysiological stress profile", *Neuropsychiatric Disease and Treatment*, vol. 14, page 435, 2018.

[25] Anna Lea Kutsch, "On the use of wearables and biofeedback interventions for people with anxiety disorders", Ph.D. dissertation, Wien, 2019.

[26] I-Mei Lin, LY Tai, and Sheng-Yu Fan, "Breathing at a rate of 5.5 breaths per minute with equal inhalation-to-exhalation ratio increases heart rate variability", *International Journal of Psychophysiology*, vol. 91, no. 3, pages 206–211, 2014.

[27] Xiao Ma, Zi-Qi Yue, Zhu-Qing Gong, Hong Zhang, Nai-Yue Duan, Yu-Tong Shi, Gao-Xia Wei, and You-Fa Li, "The effect of diaphragmatic breathing on attention, negative affect and stress in healthy adults", en, *Front Psychol*, vol. 8, page 874, Jun. 2017.

[28] Sehan Gamage Malcolm Rosario Ahmad Zainuddin, "Artificial intelligence and machine learning in the healthcare sector: A review", Jul. 2021.

[29] Rollin McCraty, Mike Atkinson, Dana Tomasino, and William Stuppy, "Analysis of twenty four heart rate variability in patients with panic disorder", *Biological Psychology*, vol. 56, pages 131–150, Jun. 2001. DOI: 10.1016/S0301-0511(01)00074-6.

[30] Ryan McGinnis, Ellen Mcginnis, Christopher Petrillo, and Matthew Price, "Mobile biofeedback therapy for the treatment of panic attacks: A pilot feasibility study", May 2019. DOI: 10.1109/BSN.2019.8771040.

[31] Paolo Melillo, Marcello Bracale, and Leandro Pecchia, "Nonlinear heart rate variability features for real-life stress detection. case study: Students under stress due to university examination", *BioMedical Engineering OnLine*, vol. 10, no. 1, page 96, 2011, ISSN: 1475-925X. DOI: 10.1186/1475-925X-10-96.

[32] Alexander Miloff, Arvid Marklund, and Per Carlbring, "The challenger app for social anxiety disorder: New advances in mobile psychological treatment", *Internet Interventions*, vol. 2, no. 4, pages 382–391, 2015, ISSN: 2214-7829. DOI: https://doi.org/10.1016/j.invent.2015.08.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214782915300014.

[33] Ranjita Misra and Michelle Mckean, "College students'academic stress and its relation to their anxiety, time management, and leisure satisfaction", *American journal of health studies*, vol. 16, pages 41–51, Jan. 2000.

[34] Mohammed H Iqbal, Abdullatif Aydin, Oliver Brunckhorst, Prokar Dasgupta, and Kamran Ahmed, "A review of wearable technology in medicine", *Journal of the Royal Society of Medicine*, vol. 109, no. 10, pages 372–380, 2016, PMID: 27729595. DOI: 10.1177/0141076816663560. eprint: https://doi.org/10.1177/0141076816663560. [Online]. Available: https://doi.org/10.1177/0141076816663560.

[35] D. Park, M. Lee, S. E. Park, J. K. Seong, and I. Youn, "Determination of Optimal Heart Rate Variability Features Based on SVM-Recursive Feature Elimination for Cumulative Stress Monitoring Using ECG Sensor", *Sensors (Basel)*, vol. 18, no. 7, 2018.

[36] Carolyn Pitceathly and Maggie Watson, "Counseling and specific psychological treatments in", *Psycho-Oncology*, page 117, 2013.

[37] Tapas Pramanik, Hari Om Sharma, Suchita Mishra, Anurag Mishra, Rajesh Prajapati, and Smriti Singh, "Immediate effect of slow pace bhastrika pranayama on blood pressure and heart rate", en, *J. Altern. Complement. Med.*, vol. 15, no. 3, pages 293–295, 2009.

[38] A. Puli and A. Kushki, "Toward automatic anxiety detection in autism: A real-time algorithm for detecting physiological arousal in the presence of motion", *IEEE Trans Biomed Eng*, vol. 67, no. 3, pages 646–657, Mar. 2020.

[39] *Pulseon ohr tracker (sp-2d-w) data sheet*, Sep. 2019.

[40] Enrique Ramón-Arbués, Vicente Gea-Caballero, José Manuel Granada-López, Raúl Juárez-Vela, Begoña Pellicer-García, and Isabel Antón-Solanas, "The prevalence of depression, anxiety and stress and their associated factors in college students", *International Journal of Environmental Research and Public Health*, vol. 17, no. 19, 2020, ISSN: 1660-4601. DOI: 10.3390/ijerph17197001. [Online]. Available: https://www.mdpi.com/1660-4601/17/19/7001.

[41] Donald Robinaugh, Mackenzie Brown, Olivia Losiewicz, Payton Jones, Luana Marques, and Amanda Baker, "Towards a precision psychiatry approach to anxiety disorders with ecological momentary assessment: The example of panic disorder", *General Psychiatry*, vol. 33, e100161, Feb. 2020. DOI: 10.1136/gpsych-2019-100161.

[42] Anna-Lena Roos, Thomas Goetz, Maike Krannich, Monika Donker, Maik Bieleke, Anna Caltabiano, and Tim Mainhard, "Control, anxiety and test performance: Self-reported and physiological indicators of anxiety as mediators", *British Journal of Educational Psychology*, 2022.

[43] Alessio Rossi, Eleonora Da Pozzo, Dario Menicagli, Chiara Tremolanti, Corrado Priami, Alina Sîrbu, David A. Clifton, Claudia Martini, and Davide Morelli, "A public dataset of 24-h multi-levels psycho-physiological responses in young healthy adults", *Data*, vol. 5, no. 4, 2020, ISSN: 2306-5729. DOI: 10.3390/data5040091. [Online]. Available: https://www.mdpi.com/2306-5729/5/4/91.

[44] Jonathan Rubin, Rui Abreu, Shane Ahern, Hoda Eldardiry, and Daniel G. Bobrow, "Time, frequency & complexity analysis for recognizing panic states from physiologic time-series", in *Proceedings of the 10th EAI International Conference on Pervasive Computing Technologies for Healthcare*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016, pages 81–88, ISBN: 9781631900518.

[45] Irwin G. Sarason and Barbara R. Sarason, *Test Anxiety*, Harold Leitenberg, Ed. Boston, MA: Springer US, 1990, pages 475–495, ISBN: 978-1-4899-2504-6. DOI: 10.1007/978-1-4899-2504-6_16. [Online]. Available: https://doi.org/10.1007/978-1-4899-2504-6_16.

[46] Hashini Senaratne, Levin Kuhlmann, Kirsten Ellis, Glenn Melvin, and Sharon Oviatt, "A multimodal dataset and evaluation for feature estimators of temporal phases of anxiety", in *Proceedings of the 2021 International Conference on Multimodal Interaction*, ser. ICMI '21, Montréal, QC, Canada: Association for Computing Machinery, 2021, ISBN: 9781450384810. DOI: 10.1145/3462244.3479900. [Online]. Available: https://doi.org/10.1145/3462244.3479900.

[47] Seyed Amir Hossein Aqajari, Emad Kasaeyan Naeini, Milad Asgari Mehrabadi, Sina Labbaf, Nikil Dutt, and Amir M. Rahmani, "Pyeda: An open-source python toolkit for pre-processing and feature extraction of electrodermal activity", *Procedia Computer Science*, vol. 184, pages 99–106, 2021, The 12th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 4th International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2021.03.021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050921006438.

[48] Hye-Sue Song and Paul M Lehrer, "The effects of specific respiratory rates on heart rate and heart rate variability", *Applied psychophysiology and biofeedback*, vol. 28, no. 1, pages 13–23, 2003.

[49] Steven R Steinhubl, Evan D Muse, and Eric J Topol, "The emerging field of mobile health", *Science translational medicine*, vol. 7, no. 283, 283rv3, 2015, ISSN: 1946-6234. DOI: 10.1126/scitranslmed.aaa3487. [Online]. Available: https://europepmc.org/articles/PMC4748838.

[50] Ashok Vegesna, Melody Tran, Michele Angelaccio, and Steve Arcona, "Remote patient monitoring via non-invasive digital technologies: A systematic review", *Telemedicine and e-Health*, vol. 23, no. 1, pages 3–17, 2017, PMID: 27116181. DOI: 10.1089/tmj.2016.0051. eprint: https://doi.org/10.1089/tmj.2016.0051. [Online]. Available: https://doi.org/10.1089/tmj.2016.0051.

[51] Okta, *Auth0, api authorization*, Last accessed 28 September 2022, 2017. [Online]. Available: https://auth0.com/access-management.

[52] World Health Organization, "Depression and other common mental disorders: Global health estimates", Technical documents, 2017, 24 p.

[53] Dominic J. C. Wilkinson, Jane M. Thompson, Gavin W. Lambert, Garry L. Jennings, Rosemary G. Schwarz, Don Jefferys, Andrea G. Turner, and Murray D. Esler, "Sympathetic activity in patients with panic disorder at rest, under laboratory mental stress, and during panic attacks", *Archives of General Psychiatry*, vol. 55, no. 6, pages 511–520, Jun. 1998, ISSN: 0003-990X. DOI: 10.1001/archpsyc.55.6.511. [Online]. Available: https://doi.org/10.1001/archpsyc.55.6.511.

[54] Da-Wei Wu, Li-Hsin Chang, Po-Chou Yang, Tzu-Yu Kuo, Dong-Lin Tsai, Huang-Chi Chen, Hui-Lan Yuan, Pei-Shih Chen, Szu-Chia Chen, and I-Mei Lin, "Anxiety is a mediator between heart rate variability and quality of life in chronic obstructive pulmonary disease", *J. Pers. Med.*, vol. 12, no. 6, page 960, 2022. DOI: 10.3390/jpm12060960.

[55] Ali K. Yetisen, Juan Leonardo Martinez-Hurtado, Barış Ünal, Ali Khademhosseini, and Haider Butt, "Wearables in medicine", *Advanced Materials*, vol. 30, no. 33, page 1 706 910, 2018. DOI: https://doi.org/10.1002/adma.201706910. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adma.201706910. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201706910.

[56] M. Zeidner and G. Matthews, *Anxiety 101*, ser. Psych 101. Springer Publishing Company, 2010, ISBN: 9780826104892. [Online]. Available: https://books.google.pt/books?id=sKMBMud9PSEC.

[57] Yali Zheng, Tracy C. H. Wong, Billy H. K. Leung, and Carmen C. Y. Poon, "Unobtrusive and multimodal wearable sensing to quantify anxiety", *IEEE Sensors Journal*, vol. 16, no. 10, pages 3689–3696, 2016. DOI: 10.1109/JSEN.2016.2539383.

[58] Jianping Zhu, Lizhen Ji, and Chengyu Liu, "Heart rate variability monitoring for emotion and disorders of emotion", *Physiological Measurement*, vol. 40, no. 6, page 064 004, 2019. DOI: 10.1088/1361-6579/ab1887. [Online]. Available: https://doi.org/10.1088/1361-6579/ab1887.

A

Anxolotl App Screens

This appendix will present most of the features implemented in the Anxolotl App, in the form of screenshots. While, we do not want to put every single possible case, we will try to present a complete overview of the capabilities that our app presents to its users.

The app was running on a Google Pixel 4a 5g device, and all the screenshots were taken using that device. The resolution of each shot is near Full HD (1920x1800px) and it is intended to be representative of a real world usage scenario.

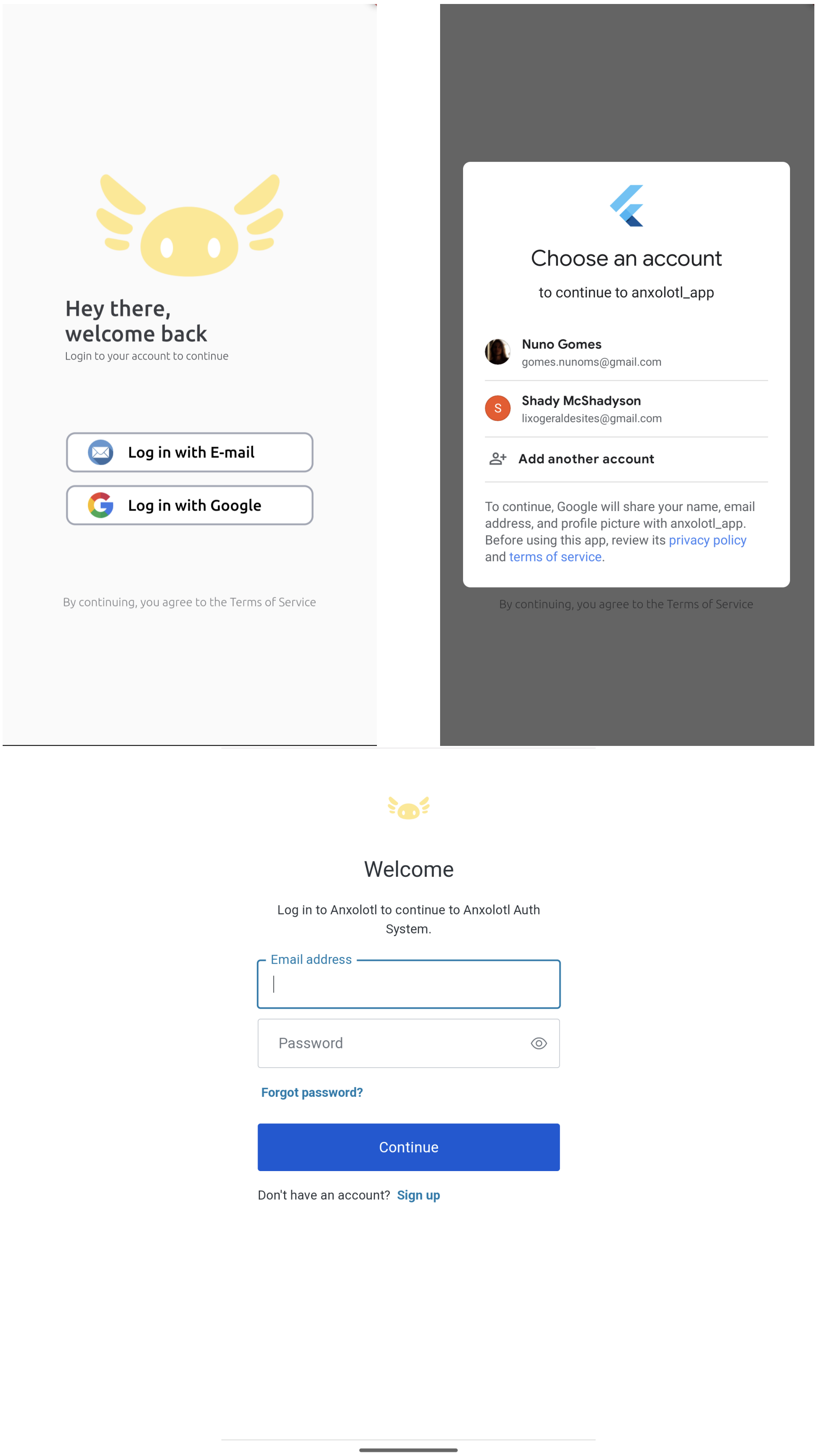

Figure A.1: Authentication section general (left), Google Authentication (right) and Auth0 authentication (down)

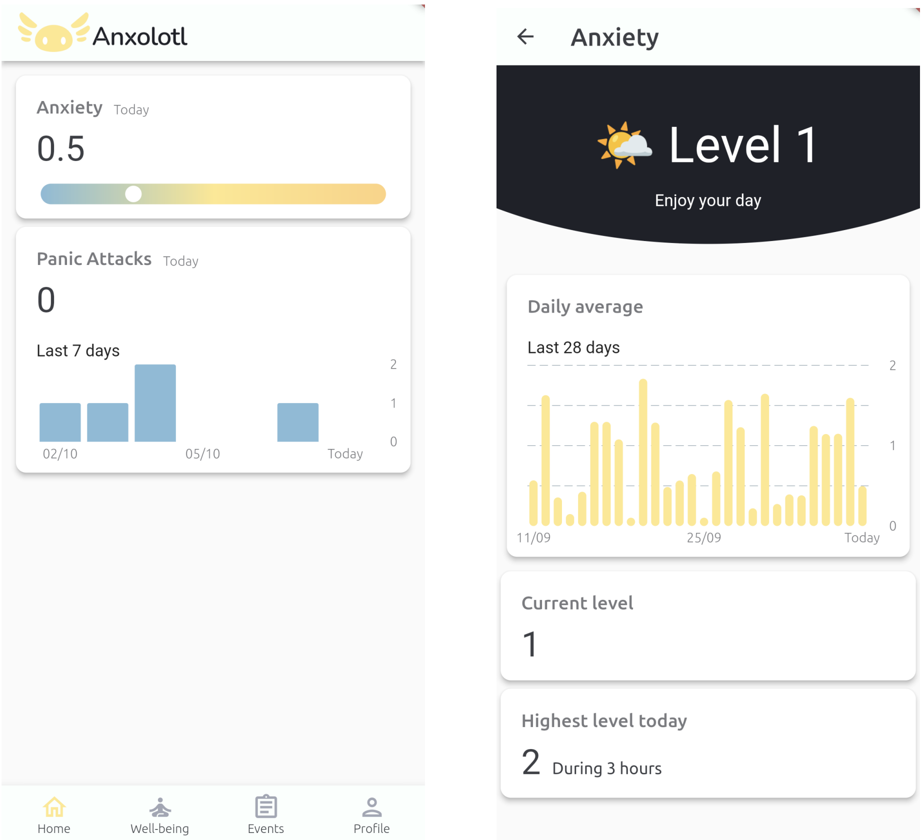

Figure A.2: Dashboard (left) and anxiety detail (right)

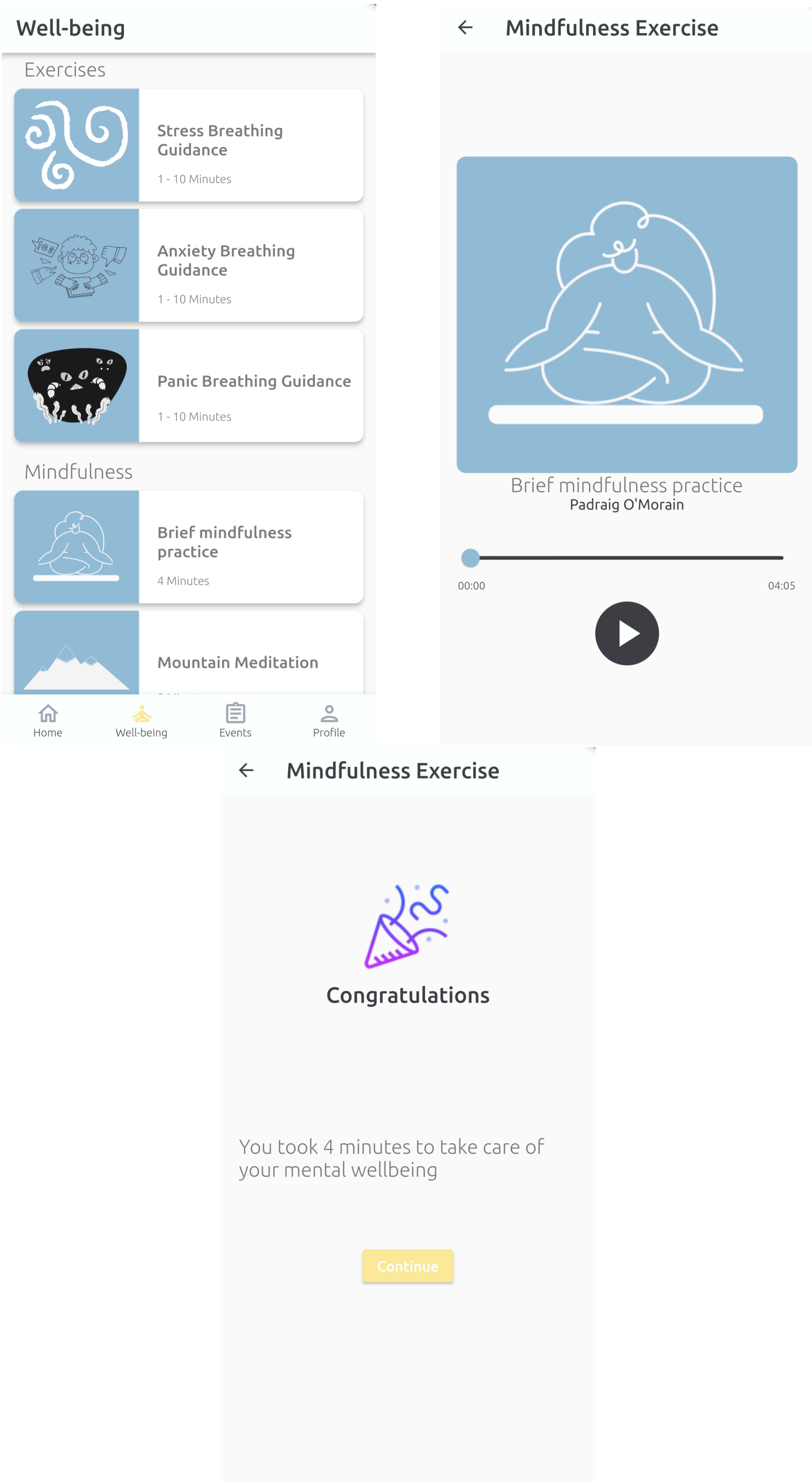

Figure A.3: Wellbeing section (left), meditation exercise (right) and meditation exercise ending (down)

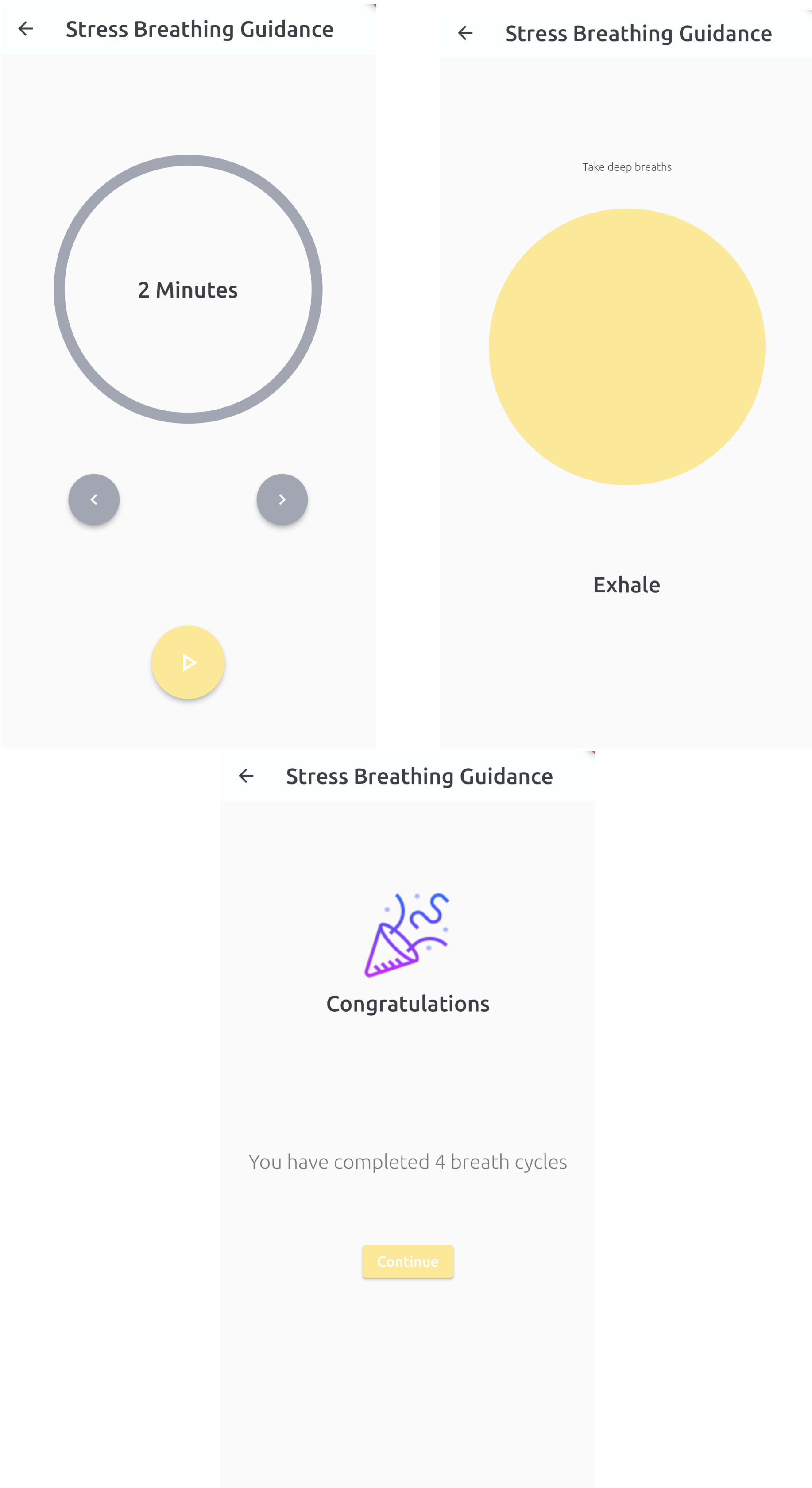

Figure A.4: Breathing exercise setup (left), breathing exercise itself (right) and exercise ending (down)

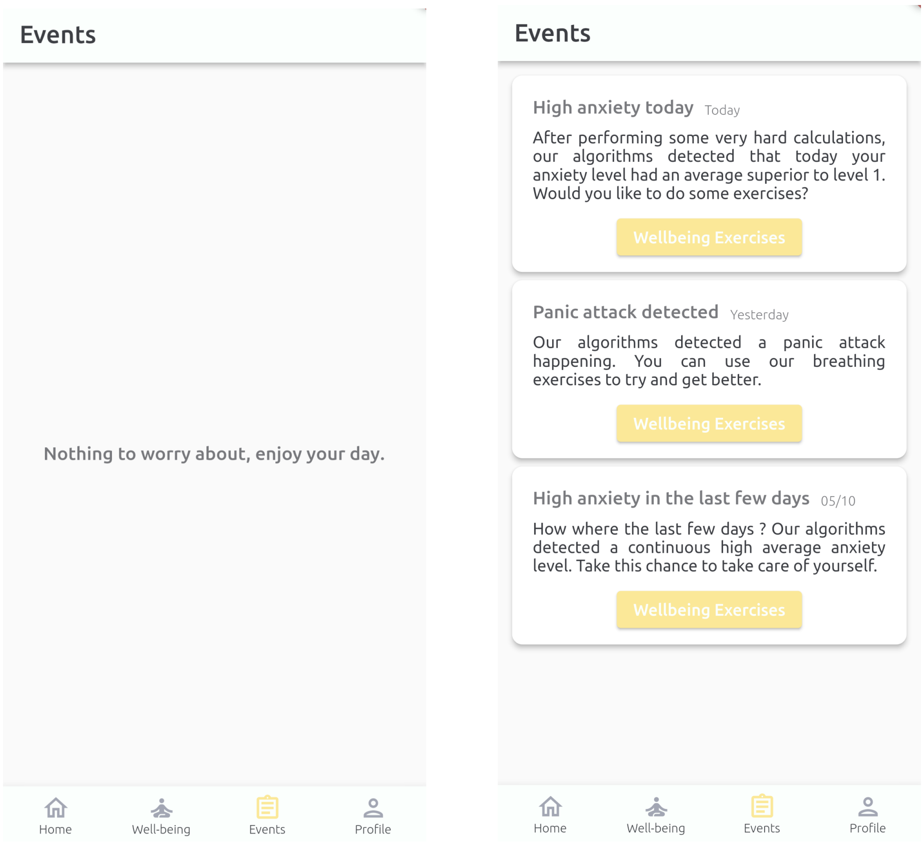

Figure A.5: Events section empty (left) and with all events (right)

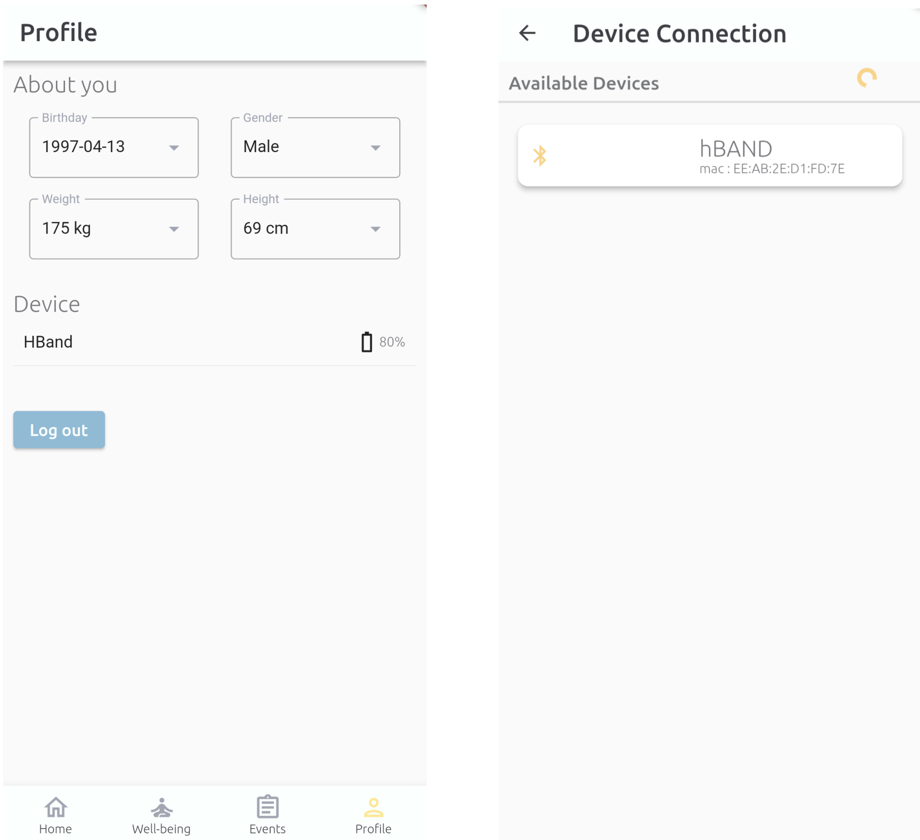

Figure A.6: Profile section (left), and Bluetooth device connection page (right)