Flow-Z: A Flow-based Adaptable Game to Maintain Optimal Challenge

Tomás Almeida e Silva Martins Alves

Thesis to obtain the Master of Science Degree in Information Systems and Computer Engineering

Supervisors: Profª Sandra Pereira Gama
Prof. Francisco António Chaves Saraiva de Melo

Examination Committee

Chairperson: Prof. Miguel Nuno Dias Alves Pupo Correia
Supervisor: Profª Sandra Pereira Gama
Member of the Committee: Dr. Daniel Simões Lopes

November 2017
Acknowledgments

Before anyone else, I would like to thank my family for their understanding, caring and support over all these years, for all the sacrifices we have been through, for always being there for me through thick and thin and without whom this project would not be possible.

I would also like to acknowledge my dissertation supervisors Prof. Sandra Gama and Prof. Francisco Melo. I am proud of the job I developed for over a year and they are part of its pillars. They always showed me the path and motivation to keep this work going until the end with their insight, support and sharing of knowledge.

I want to thank Catarina Lima for being my safe port, for all the patience and dedication, for taking my head out of the books and relax a little (otherwise I would be mad by now), for everything. Also, a special thanks my friends and colleagues Inês Santos, Luís Fonseca, Miguel Santos and Rodrigo Veríssimo for their time, knowledge, dedication and friendship. You’re simply the best. I also thank to everyone that participated in my testing phases.

A big thank you to the Visualization and Intelligent Multimodal Interfaces (VIMMI) group, specially to Prof. Joaquim Jorge, by the way they welcomed my project, to the Instituto de Biofísica e Engenharia Biomédica (IBEB), specially to Prof. Hugo Ferreira for launching me in this project, to the PLUX wireless biosignals S.A. team, specially to Prof. Hugo Silva for the fast and enlightening answers, and to Fundação para a Ciência e a Tecnologia (FCT). This work was supported by national funds through FCT with reference UID/CEC/50021/2013.

Last but not least, I want to acknowledge all my friends and colleagues that helped me grow as a person and were always there for me during the good and bad times in my life.

To each and every one of you – Thank you.
Abstract

Often video games fail to attract a wider range of consumers because people become uninvolved when they fail to meet the in-game difficulty. This dissertation addresses the problem of the in-game difficulty not being correctly adapted to the gamers, leading to their uninterest for not having their skills balanced with the challenge of the game. This balance is one of the conditions that lead people to flow, which is the mental state associated with optimal enjoyment of an activity. In our work, we study if flow may be relevant for gameplay adaptability and may offer a better gaming experience, since it provides a better enjoyment of an activity. We created a hypothesis to verify if a game that adapts its parameters to a representation of the mental state of the player following the flow theory can provide a better gaming experience compared to a game that adapts to their performance. We developed a first-person shooter video game that adapts its in-game difficulty and environmental settings based on a representation of their mental state to keep a balance between the skills of the player and the challenge of the game. The mental state of the player is measured with their physiological signals, namely the heart rate and the beta band of the brainwaves, and we distinguish the mental state of the player with an accuracy of 87%. We also conducted an evaluation using self-perceived flow and in-game scores as metrics to compare the mental state-based adaptability with a performance-based version. Results show that the latter provided a better gaming experience.

Keywords

Video games, optimal experience, flow, performance, adaptable gameplay, psychophysiology
Resumo

Muitas vezes, os videojogos não conseguem atrair um leque mais amplo de consumidores porque as pessoas não se envolvem quando não se conseguem enquadrar na dificuldade no jogo. Esta dissertação aborda o facto da dificuldade dos jogos não ser convenientemente adaptada ao jogador, levando-o ao desinteresse dado que não existe um balanço entre as suas capacidades e o desafio do jogo. Este balanço é uma das condições que levam ao estado de flow, o qual é o estado mental associado ao aproveitamento ótimo de uma atividade. Neste trabalho, verificamos se o flow pode ser um fator importante na adaptabilidade do gameplay e oferecer uma melhor experiência de jogo, dado que proporciona um melhor aproveitamento dum a atividade. Criamos a hipótese de que um jogo que se adapta a uma representação do estado mental do jogador segundo a teoria do flow proporciona uma melhor experiência de jogo, dado que proporciona um melhor aproveitamento dum a atividade. Desenvolvemos um jogo de vídeo de first-person shooter que adapta a sua dificuldade e as configurações ambientais do jogo dependendo da representação do estado mental do jogador de forma a manter um equilíbrio entre as capacidades do jogador e o desafio do jogo. Medimos o estado mental do jogador com os seus sinais fisiológicos, nomeadamente a frequência cardíaca e a banda beta das ondas cerebrais, e distinguimos o estado mental do jogador com uma precisão de 87%. Também realizamos uma avaliação usando o flow percecionado pelo jogador e a sua pontuação final como métricas para comparar a adaptabilidade baseada no estado mental com uma versão baseada no desempenho do jogador. Os resultados indicam que esta última oferece uma melhor experiência de jogo.

Palavras Chave

Jogos de vídeo, experiência ótima, flow, adaptação de jogabilidade, psicofisiologia
Contents

1 Introduction 1
  1.1 Contributions .................................................. 4
  1.2 Document Structure ........................................... 5

2 Fundamentals on Flow Theory 7
  2.1 Flow Theory ..................................................... 9
  2.2 Physiology of Flow ............................................ 12
  2.3 Flow in experimental settings ................................. 13

3 Related Work 15
  3.1 First-Person Shooters .......................................... 17
  3.2 Flow and Gameplay ............................................ 18
  3.3 Adaptable Gameplay ........................................... 20
  3.4 Classification Algorithms for Mental State Recognition 21
  3.5 Discussion ..................................................... 22

4 System Overview 25
  4.1 Game ......................................................... 27
    4.1.1 Game Choice ............................................... 27
    4.1.2 Base Game Design ........................................ 29
    4.1.3 Adaptable Game Parameters .............................. 32
    4.1.4 Inducement Versions Design .............................. 34
    4.1.5 User Studies ............................................. 35
      4.1.5.1 Participants ........................................... 36
      4.1.5.2 User Evaluation ...................................... 36
      4.1.5.3 Apparatus ............................................. 40
      4.1.5.4 Procedure ............................................ 41
    4.1.6 Analyzing Data ........................................... 44
      4.1.6.1 PANAS ................................................. 44
      4.1.6.2 Adapted GEQ ......................................... 47
4.1.7 Discussion ................................................. 51
4.2 Classification Framework ...................................... 52
  4.2.1 Choosing the Physiological Measures ...................... 52
  4.2.2 Classification Algorithm .................................. 55
    4.2.2.1 Choosing Features .................................. 56
    4.2.2.2 Choosing a Classifier ................................ 60
    4.2.2.3 Discussion ......................................... 65
  4.2.3 Architecture ............................................. 66
  4.2.4 Discussion ............................................. 66
4.3 Controller Framework .......................................... 66
  4.3.1 State Machine ........................................... 67
  4.3.2 Environmental Settings ................................... 68
  4.3.3 Difficulty Settings ....................................... 69
  4.3.4 Discussion ............................................. 70
5 Validation .......................................................... 71
  5.1 Architecture ................................................ 73
  5.2 Method ..................................................... 74
    5.2.1 Participants ........................................... 74
    5.2.2 User Evaluation ......................................... 74
    5.2.3 Apparatus .............................................. 75
    5.2.4 Procedure ............................................. 75
  5.3 Results ...................................................... 76
    5.3.1 PANAS ................................................ 77
    5.3.2 Flow ................................................... 78
    5.3.3 In-Game Scores ......................................... 79
  5.4 Discussion ................................................ 81
6 Conclusions and Future Work .................................... 83
List of Figures

2.1 Flow-channel model adapted from Csikszentmihalyi [1]. ........................................ 10
2.2 Four-channel flow model presented by Nacke and Lindley [13]. ................................. 10
2.3 Flow model proposed by Chen et al. [15]. ................................................................. 10
2.4 EFM proposed by Massimini and Carli [16]. ............................................................... 10

4.1 User and the three core components of the solution. .................................................. 27
4.2 Representative picture of the Nightmares Survival Shooter tutorial in Unity Asset Store. .. 28
4.3 Zombunny enemy in-game. ............................................................................................ 28
4.4 Zombear enemy in-game. ............................................................................................... 28
4.5 Hellephant enemy in-game. ......................................................................................... 28
4.6 Game world seen from a corner of the room. ............................................................... 29
4.7 Rocket object model. .................................................................................................... 30
4.8 Explosion in-game. ....................................................................................................... 30
4.9 Health pick-up in-game. ............................................................................................... 31
4.10 Machine gun ammo pick-up in-game. ......................................................................... 31
4.11 Rocket launcher ammo pick-up in-game. ..................................................................... 31
4.12 Updated GUI. .............................................................................................................. 32
4.13 Sketch of the mental states induced by the game as a function of the enemy parameters health and speed. ...................................................................................... 33
4.14 Sketch of the mental states induced by the game as a function of the enemy parameters health and spawn time. .............................................................................. 33
4.15 Sketch of the mental states induced by the game as a function of the enemy parameters speed and spawn time. ................................................................. 34
4.16 Positive and Negative Affect Schedule (PANAS) scale from the user form. .............. 38
4.17 Adapted Game Engagement Questionnaire (GEQ) scale from the user form. .............. 39
4.18 Main board of the (r)evolution Plugged Kit from Bitalino with three sensor cables connected to it. ......................................................................................... 41
List of Tables

2.1 Brainwave Bands [25]. ................................................................. 13

3.1 Eight elements of flow and corresponding gameplay elements [54]. .............. 19

4.1 Version name, the mental state it is intended to induce and the general characteristics of each game version. ................................................................. 35

4.2 Stats differences between game versions. ........................................... 36

4.3 Asymptotic significance (2-tailed) values between alpha and theta bands and HR for each pair of versions. ................................................................. 55

4.4 Identifiers for each feature. ............................................................. 58

4.5 Parameters of the decision tree with highest accuracy. .......................... 61

4.6 Confusion matrix of decision tree with test set. .................................... 61

4.7 Classification report of decision tree with test set. ............................... 61

4.8 Importance of each feature of decision tree with test set. ........................ 62

4.9 Parameters of the random forest with highest accuracy. ........................ 62

4.10 Confusion matrix of random forest with test set. ................................. 62

4.11 Classification report of random forest with test set. ............................ 63

4.12 Importance of each feature of random forest with test set. ...................... 63

4.13 Parameters of the support vector machine with highest accuracy. ............ 63

4.14 Confusion matrix of support vector machine with test set. ..................... 64

4.15 Classification report of support vector machine with test set. .................. 64

4.16 Parameters of the multilayer perceptron with highest accuracy. ............... 64

4.17 Confusion matrix of multilayer perceptron with test set. ........................ 64

4.18 Classification report of multilayer perceptron with test set. .................... 65

4.19 Comparison of the classifiers with average test set results. ..................... 65

4.20 Stats differences between levels. ...................................................... 70
Acronyms

ANOVA  Analysis of Variance
ANS  Autonomic Nervous System
AOE  Area of Effect
BCI  Brain-Computer Interface
BVP  Blood Volume Pulse
DT  Decision Tree
EDA  Electrodermal Activity
EEG  Electroencephalography
EFM  Experience Fluctuation Model
EMG  Electromyogram
FFT  Fast Fourier Transform
fMRI  Functional Magnetic Resonance Imaging
FPS  First-Person Shooter
GEQ  Game Engagement Questionnaire
GSR  Galvanic Skin Response
GUI  Graphical User Interface
HF-HRV  High-frequency Band
HR  Heart Rate
HRV  Heart Rate Variability
IAPS  International Affective Picture System
IBI  Interbeat Interval
IQR  Interquartile Range
LDA  Linear Discriminant Analysis
LF-HRV  Low-frequency Band
MLP  Multilayer Perceptron
OXY  Normalized Oxygenation Changes
PANAS  Positive and Negative Affect Schedule
RDA  Abdominal Respiratory Depth
RDT  Thoracic Respiratory Depth
RF  Random Forest
RR  Respiratory Rate
SVM  Support Vector Machine
Introduction
Nowadays, video games are part of our modern society for both entertainment and education. According to *The 2016 Essential Facts About the Computer and Video Game Industry*¹ released by the Entertainment Software Association (ESA) in April 2016, 63% of U.S. households are home to at least one person who plays video games regularly (3 hours or more per week) and there is an average of 1.7 gamers in each game-playing U.S. household. With so many players in the market, in order to achieve a certain level of success, games must appeal to a range of consumers as wide as possible.

Since traditional game design offers a discrete scope of difficulty, players may end up feeling alienated for not falling on the discriminatory set of in-game difficulties, *i.e.* some players may find a game that is either too easy or too hard for them. This dissertation addresses the problem of the in-game difficulty not being correctly adapted to the gamers, leading to their uninterest for not having their skills paired with the challenge of the game.

The balance between the challenge of an activity and the skills of an individual is one of the conditions that lead people to flow [1]. This concept is associated to the mental state that people feel when they are completely engaged in an activity and have an optimal experience while performing it. In order to avoid having players thinking that a game is either too easy or too hard, we must keep a balance between the challenge of the game and the skills of the player. By varying the settings of the game, we keep this balance through adaptation of the in-game difficulty to the expertise of the player. Therefore, a game which can provide an adequate challenge level and dynamically adapt it in real time to the player will provide a better gaming experience.

Since everyone has a different gaming experience, adaptable gameplay should address differently each individual rather than treating the gaming community as a whole group. Biofeedback comes as an answer to detect psychophysiology features and is a useful tool to measure those indicators so that it is possible to calculate the appropriate measures to be taken accordingly to one’s affective state. This way, user oriented gaming experience can be provided to the gamer.

There are already some games with adaptable gameplay, such as a method of difficulty adaptation related to user anxiety [2], user emotions [3], user frustration [4] or user performance [5]. A game that adapts its gameplay difficulty and environment interactions allows a better gaming experience, using the former to balance the challenge of the game and the skills of the player and the latter to create a greater sense of engagement in the player. However, there are not any games that adapt their gameplay to the user’s flow and its complementary mental states beyond game difficulty. This led us to state our research goal as:

*If flow provides a better enjoyment of an experience, then flow may be relevant for gameplay adaptability and may offer an enhanced gaming experience.*

Having the goal stated, we designed and created a game with an adaptability component. Our

game is an endless zombie waves First-Person Shooter (FPS) where players have to play to score as many points as they can. Since we cannot control the level of skill development of the player, we have to change the challenge of the game depending on the mental state of the player. The adaptable parameters are able to change the in-game difficulty and the environment, thus allowing us to provide the best answer to offer a better gaming experience.

The characteristic that categorizes our Brain-Computer Interface (BCI) as a passive one is the fact that our classification framework derives its outputs from arbitrary brain activity arising without the purpose of voluntary control, to enrich human–machine interaction with implicit information about the current user state. In this way, we complement the conscious manual input from the user to the game with a secondary channel, influencing and enriching the ongoing primary interaction with implicit user information [6]. More specifically, while the user is playing the game, the classification framework is reading the player's physiological signals and outputting a representation of their mental state to the controller framework. The controller framework is responsible for changing the game parameters depending on the representation of the mental state of the player, accordingly to the flow theory.

In order to verify our goal, we need to compare our adaptability component based on flow against another adaptability that does not depend on the mental state of the player. Therefore, we created the following hypothesis: **gamers have a better gaming experience playing a game that adapts to the representation of their mental state by keeping them in flow compared to a game that adapts to their performance.** To validate our hypothesis, we created and ran two prototypes in user testing: one adapts to the representation of the mental state of the user and the other to their performance. We used self-perceived flow and in-game performance as metrics to evaluate the prototypes. We defined that to prove our hypothesis our objectives are that players have a higher performance and a higher self-perceived flow when they are playing a game that adapts to the representation of their mental state compared to playing a game which gameplay adapts to their performance.

### 1.1 Contributions

This dissertation aims to provide useful inputs about how flow can be relevant for gameplay adaptability to provide the best possible gaming experience. More precisely, we contribute with:

- A game that allows adaptation of its parameters and environment;
- A classification framework that detects the mental state of the player through biofeedback measures;
- A controller framework that adapts a game based on the mental state of the player to maintain full engagement from the user; and
• A set of physiological data recorded from participants as they performed our tests.

We also wrote and submitted an article for the journal Computers in Human Behavior from Elsevier. The article’s focus is the assessment of flow in FPS games with a choice of physiological measures.

1.2 Document Structure

This document is organized as follows. Chapter 2 sets the background of this work explaining the concept of flow, along with the flow model evolution. In order to measure flow and to understand when the player is in this state, physiology of the flow is also introduced in this section in a matter of explaining which physiological indicators are more adequate and effective. This chapter ends with several examples of flow assessment and inducement studies.

Chapter 3 starts by presenting the relation between flow and gameplay and how we can exploit that relation to achieve the greatest possible user engagement. There are also various studies explaining how the authors chose to adapt the gameplay of their games to elicit affective states. Also, this chapter explains how a great range of machine learning and pattern recognition algorithms have been used in the past to assess differences between the affective states of users. A discussion on all the presented adaptable gameplay techniques and classification algorithms ends this chapter.

Next, chapter 4 describes an overview on the necessary elements to have a functional game adapting to the user’s mental state. These components are the adaptable game and the classification and controller frameworks. Each component is fully described alongside all the development work.

The final solution is then introduced in chapter 5. We present the final architecture and the validation of our prototype. Results are then discussed alongside the objectives previously presented. The document ends with a conclusion on all the developed work and a discussion on future work.
Fundamentals on Flow Theory

Contents

2.1 Flow Theory ...................................................... 9
2.2 Physiology of Flow .......................................... 12
2.3 Flow in experimental settings ............................. 13
The core concept to understand and develop our system is flow. Csikszentmihalyi defined flow as a mental state that people feel when they are completely engaged in an activity [1]. Psychologically, the heart of the flow theory [1] is akin to cognitive theories of emotions [7, 8, 9]. These theories back the idea that cognitive evaluations of events in the world are fundamental parts of emotions. The concept of flow has been studied for several years and the flow model, i.e. a two-dimensional graph with the player’s skills and the game’s challenge as axis, has evolved due to the results of those studies. Physiological indicators have been used as a substitute of the traditional questionnaires to access user’s flow, since the former can determinate user flow without having the user consciously processing the experience. In this section, we give an overview regarding how the flow theory has evolved along the years. This section also presents the physiology factors of the flow state and ends with several studies on flow assessment and induction.

2.1 Flow Theory

Csikszentmihalyi [10] addresses the feeling of deep engagement as state of “flow”: "Flow tends to occur when a person’s skills are fully involved in overcoming a challenge that is just about manageable. ... When goals are clear, feedback relevant, and challenges and skills are in balance, attention becomes ordered and fully invested." Flow is reported as a state of being similar across all types of people, in a sense that all people recognize flow when it is explained to them [11].

After studying the intrinsically motivated behavior of artists, chess players, musicians and sports players, Csikszentmihalyi introduced the flow model [1], shown in Figure 2.1. That group felt rewarded by executing actions per se, meaning that they were experiencing high enjoyment and fulfillment in the activity itself rather than its goals. Csikszentmihalyi continued his work to study people during their ordinary lives, using the experience sampling method. Based on his findings, Csikszentmihalyi defined the following conditions for flow:

- Perceived challenges of the activity match and stretch the capabilities of the individual, thus producing an experience of being fully engaged in the task and acting on the height of their skills [11, 12]; and

- The goals of the activity are explicit and reachable, and one receives instant feedback for their progress on the activity [12].

Test data only partially corroborated Csikszentmihalyi’s hypothesis. Subjects experienced flow when they first met a task with a high balance between skills and challenges; yet, when challenges and skills were too low at the start or when a task had to be repeated frequently, apathy rather than engagement was reported, contradicting the Flow-channel model proposed by Csikszentmihalyi. Nacke and Lindley [13] presented a two-dimensional four-channel model of flow based on Csikszentmihalyi [1] and Ellis
et al. [14] which incorporates the apathy state and is used most frequently for describing games and gameplay experience. This model is illustrated in Figure 2.2.

Further research led the creation of two more flow models that are worth mentioning. Chen [15] also updated the flow model, as shown in Figure 2.3, by proposing different "flow zones" for hardcore and novice players and an optimal intersection, within which the gaming experience concentrates towards an optimal combination of game challenges and player skills.

Another researcher on optimal experience proposed the Experience Fluctuation Model (EFM) [16], shown in Figure 2.4, resolving in this way the difference between test data and the original flow model.
The EFM divided the plane into eight parts and test data from a study of Italian school-goers confirmed its accuracy [17]. The graph’s midpoint is a person’s average challenges and skills for an activity. The factorization of flow into properties and the relative precision of the EFM is what made flow such an extensive and valuable tool for studying computer mediated experiences.

In spite of the existence of the models, there must be some elements to help assessing the flow state and the conditions for experiencing it. Csikszentmihalyi [11] defined nine dimensions of flow which are common to the experience and are used as a basis for several measurement instruments (e.g., [18, 19]):

- A balance between the challenge of the task and skills of the individual;
- A merging of action and awareness, i.e. one performs the activity almost “automatically”;
- Clearly perceived goals;
- Unambiguous feedback;
- Focusing on the task at hand;
- A sense of control of the activity;
- A loss of self-consciousness or a reduced awareness of self;
- Time transformation, i.e. sense of time becomes distorted; and
- An autotelic, intrinsically rewarding experience, implying that the activity in itself is a reason for performing it instead of any external objectives [11, 12].

Recently, Cheng et al. [20] conceived a four-stage flow formation model which encapsulates these dimensions. Stage one encompasses media contents variables, namely interactivity, involvement and vividness. Stage two has the premises and perceptions before into a flow state (challenge, focused attention and skill), while stage three is the flow state containing the flow itself and telepresence. Finally, the forth stage has the consequences of flow, specifically loyalty and positive affect.

Our study focuses on the flow model presented by Nacke and Lindley [13]. As further discussed in Section 4.2, we address different representations of three mental states: flow and its two complementary mental states of anxiety and boredom. Regarding the apathy state, players will be able to play a sandbox version of the game and develop their initial skills. In that version, players can take as much time as they need to understand the basic concepts of the game and all the possible in-game interactions. The participants only stopped playing when they were sure that they understood how the game mechanics work and after combat training against some enemies. This takes away the need to address apathy, since we assume that participants will not have skills too low for the task.
2.2 Physiology of Flow

Questionnaires and interviews are the most used methods to measure flow experience. Yet, since they are carried out after the activity, they are subjective because the user is able to reflect on what they experienced instead of providing unprocessed input. Since flow happens when the user is fully engaged in the task at hand, there is no interference of a conscious control system. In order to prevent the reflection on one’s own experience, one solution is to use physiological flow indicators which are objective and can be measured during the task without interrupting the user [21].

Bian et al. [22] concluded in their study that Heart Rate (HR), Interbeat Interval (IBI), Heart Rate Variability (HRV), Low-frequency Band (LF-HRV), High-frequency Band (HF-HRV) and Respiratory Rate (RR) are all efficient flow indicators. They also discovered that Autonomic Nervous System (ANS) activation is approximately associated with the flow experience and that the function between flow and HRV, LF-HRV and HF-HRV has an inverted-u-shape. In particular, HR, HRV, RR can positively predict flow experience while IBI can negatively predict flow. Adding to it, Tozman et al. [23] found in their results that there was a negative linear relation between LF-HRV and flow when the conditions of the latter were met comparing to anxiety conditions. Harmat et al. [5] also studied how HR, LF-HRV, HF-HRV, RR, Thoracic Respiratory Depth (RDT), Abdominal Respiratory Depth (RDA) and Normalized Oxygenation Changes (OXY) can be used to distinguish flow from the complementary mental states.

Another physiological measure used to assess flow is the Galvanic Skin Response (GSR). Lang et al. [24] discovered that the mean value of the GSR is related to the level of arousal. A GSR sensor measures the differences in the sweat and gland activity on the surface of the skin and the resistance of the skin decreases as the sudation increases, which often happens when one experiences emotions such as stress or surprise. GSR has been used by in emotion [3] and flow [25] assessment studies.

Concerning brainwave activity, various studies have tried to label the most effective indicators of flow when the user is playing video games. Berta et al. [25] subdivided brain frequencies in bands as represented in Table 2.1 and concluded that the most informative bands are those around alpha and beta for distinguishing among gaming conditions and mid beta to distinguish gaming from other activities. They also concluded that the difference in shape for alpha, low beta and mid beta bands can distinguish the flow state from the non-flow states boredom and frustration.

Chanel et al. [3] found differences in the beta and theta bands to distinguish between the player trying an easy, medium or hard difficulty. Nacke et al. [26] also studied the repercussion of level design in player-game interaction on brainwave activity measured with an Electroencephalography (EEG). They reported that immersion-level design evokes more activity in the theta band.

Due to technical limitations, our study only uses HR, GSR and EEG (alpha, beta and theta bands) to assess the player’s mental state. These signals were also selected because they can be measured non-
Table 2.1: Brainwave Bands [25].

<table>
<thead>
<tr>
<th>Band</th>
<th>Wavelengths (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>1-4</td>
</tr>
<tr>
<td>Theta</td>
<td>4-8</td>
</tr>
<tr>
<td>Alpha</td>
<td>8-12</td>
</tr>
<tr>
<td>Low Beta</td>
<td>12-15</td>
</tr>
<tr>
<td>Mid Beta</td>
<td>15-20</td>
</tr>
<tr>
<td>High Beta</td>
<td>20-30</td>
</tr>
<tr>
<td>Gamma</td>
<td>30-32</td>
</tr>
</tbody>
</table>

invasively and are rather resistant to movement artifacts. These artifacts are created by the participant whenever they move and the electrodes attached to them change the pressure they apply to the skin. Since, physiological activity has a logical topographic field of distribution, artifacts lead to an illogical distribution that defies the principles of localization.

In order to detect flow or the complementary mental states, we need to understand how HR, GSR and brainwaves relate to human biology. First, regarding HR, the sympathetic nervous system activity increases HR, whereas parasympathetic nervous system activity is responsible for its decrease. Anxiety usually leaves users with a racing heart, contrary to the boredom and flow mental states.

Second, the production of sweat in the eccrine sweat glands is entirely controlled by the human sympathetic nervous system. As previously mentioned, the resistance of the skin decreases as the sudation increases. Thus, an increase of arousal levels, which leads to an increase of electrodermal activity and, consequently, an increase of sudation, will manage to decrease the resistance of the skin. When one is anxious, the production of sweat increases commonly resulting in sweaty palms, for example.

Lastly, the brain wave bands also have different mental states associated with their activity. Alpha band is a marker of relaxed, yet not conscious, drowsy and tranquil mental state, low beta band indicates a relaxed, but focused and unified state and the theta band is a flag of relaxed, yet dreamy, with high alertness, meditative and sleepy state. Anxiety leaves one with a feeling of being “on-edge”. On the contrary, bored people feel tiredness and weariness caused by not being able to engage in an activity. Lastly, the flow state, as mentioned in Section 2.1, people are focused, relaxed and with a meditative reduced awareness of self.

2.3 Flow in experimental settings

Flow regarding human behavior and computers has most notably been studied regarding user experience and as a means to explain user engagement.

Flow assessment was examined in some fields of research, such as are communication [27], game-based learning [19, 28], human–computer interaction [29, 30], instant messaging [31], marketing [32, 33], mobile technologies [34], online shopping [35, 36] and websites in general [37]. Besides flow
assessment, flow induction has also been studied. In these studies [e.g., \cite{38, 39}] participants are usually asked to engage in a task. More precisely in the video games field, other studies \cite{13, 25} created different versions of a specific video game to lead users to certain mental states or feelings.

Keller et al. \cite{40} stated that finding the match between the skill level of the user and the challenge of the activity is one of the greatest difficulties regarding the inducement of flow. When users are given self-involvement and if their success in the activity is justified \cite{41} or when they feel committed to accomplishing task \cite{42}, people will show more effort to fulfill that task.

Several studies usually compare the flow condition (an equilibrium between skills and challenge) against anxiety (challenge exceeds skills) and boredom (skills exceed challenge) conditions. Nevertheless, inducing either anxiety or boredom is difficult since different people have different interest and stress limits.

Thus, in our study, the game design was carefully chosen so that participants would be more self-involved while playing the game versions, which led to adequate states of anxiety, boredom, engagement and frustration, as it is further explained in Section 4.1.4.
Related Work

Contents

3.1 First-Person Shooters ........................................... 17
3.2 Flow and Gameplay ............................................. 18
3.3 Adaptable Gameplay ............................................ 20
3.4 Classification Algorithms for Mental State Recognition .... 21
3.5 Discussion ....................................................... 22
This chapter starts by addressing recent studies regarding FPS games and their impact for our study. It also covers the relation between flow and gameplay which has already been study for several years. This chapter also goes over relevant work regarding how flow and gameplay are connected and how the latter can be adapted to induct the former. Finally, this chapter includes studies regarding classification algorithms that have been used to classify mental states and a brief discussion on the impact of the related studies on our own.

### 3.1 First-Person Shooters

FPS is one of the most popular contemporary genre of digital games, given that its action gameplay leads players to an arousal and engagement state. Several studies [13, 43, 44, 45, 46, 47] address FPS games and how players interact with them. One interesting topic is the relation between the player experience and how it relates to their biofeedback. Knowing how the user experience is related to psychophysiological measures has a great potential in the game industry, since it allows designers to address specific affective states and adapt the gameplay to how they want players to feel. Drachen et al. [43] studied if HR and Electrodermal Activity (EDA) could be used to make inferences on the arousal level of the player as they played three different FPS games: Prey\(^1\), Doom 3\(^2\) and Bioshock\(^3\). They found that psychophysiological arousal and self-reported gameplay experience have a consistent significant correlation across the three games, yet, the covariance between psychophysiological measures and self-reports varies between the two measures. Takatalo et al. [44] also address the level of user arousal while playing a FPS with EDA.

Nacke et al. [45] also studied if physiological measures such as Electromyogram (EMG) and EDA could provide a reliable measure of the affective user experience. They used a modification of Half-Life 2\(^4\) in which sound (on/off) and music (on/off) were manipulated. Participants also filled a GEQ [48] after playing the game. Their results indicated no main or interaction effects of sound or music on EMG and EDA, yet they found a significant main effect of sound on all GEQ dimensions. Additionally, they found an interaction effect of sound and music on GEQ dimension tension and flow indicates an important relationship of sound and music for gameplay experience. This is an interesting point regarding flow for our work, and we decided to take special attention to the sound effects and background music of our FPS.

Other interesting topic is that FPS games require players to develop a high cognitive-control, meaning that they have a flexible mindset to rapidly react to fast moving visual and auditory stimuli, and to switch back and forth between different subtasks. Colzato et al. [46] compared video game players against

---

\(^1\)en.wikipedia.org/wiki/Prey_(2017_video_game)  
\(^2\)en.wikipedia.org/wiki/Doom_3  
\(^3\)en.wikipedia.org/wiki/BioShock  
\(^4\)en.wikipedia.org/wiki/Half-Life_2
participants with little to no video game experience and found that the former showed greater cognitive flexibility than the latter. With this in mind, we have to choose carefully participants that have a FPS gaming periodicity of, at least, once a month so that they are used to the cognitive workload of this genre.

In another approach, Quax et al. [47] found that the players’ perception of the quality of the game depends on the size of the delay the network introduces. More specifically, there are indications that from 60 ms round-trip delay on, the player experiences the impairment as disturbing and it has a negative influence on the affected players’ perceived game quality and performance. In our testing phase, we are dependent of the network in the laboratory. Since we are not considering a continuous transmission of data, i.e. the game decides on how to adapt the parameters at a specific time interval, users do not have a perception of whether or not the game is transmitting information. Therefore, user perception of the game is not affected.

3.2 Flow and Gameplay

Flow and gameplay are generally linked in contexts where the user finds a familiar formal structure of the game, yet novel content created by the system design and user choice within the system [49]. Thus, the user weights their familiarity and experiences of the formal structure to conclude if the challenge they are facing is both achievable and desirable. From this flow experience is a positive affective response carrying enjoyment and satisfaction that leads one to both reflect positively on the experience and, typically, want to re-engage with it again [50].

Sweetser and Wyeth [51] created GameFlow, a succinct model of enjoyment in game evaluation which is a combination of varied heuristics and user experience. They treated player enjoyment as flow and work on the assumption that it should be a standard for designing and evaluating games, based on Csikszentmihalyi’s dimensions of flow and the added factor of player interaction. The designer is advised to allow the player to be able to concentrate on the game, to keep the user enjoyment by showing clear goals, giving autonomy (they are able to feel a sense of control over their actions in the game) and feedback at proper time to the player. Proper narrative that leads to the player’s pleasure is also advised given the heuristics of immersion (players sense deep, yet effortless involvement), challenge and allow social interaction. Ultimately, the game design should be able to provide player skill development and mastery needed for the player skills factor.

Csikszentmihalyi [11] stated that a key argument why players experience flow is an adequate level of challenge, yet this challenge does not necessarily lead to flow. The balance between one’s skills and the difficulties of the task at hand is a catalyst for flow, therefore a player will feel bored or anxious if their skills exceed or fall short of the challenge they are facing. Sweetser and Wyeth [51] declared that players’ perceived skills should match the challenge given in the game, thus the game must display a
right balance between challenge and ability so that it facilitates and maintains flow during gameplay, keeping players inside the Flow Zone [52]. This is also true for ‘mini-games’, in a sense that players achieve quick outcomes, and for complex games, since there are goals and sub-goals. The game should vary the level of difficulty at an appropriate pace by providing adequate challenges easily matched by the player’s skills [53].

In Table 3.1, Cowley et al. [54] map eight of the flow elements and respective gameplay elements. There is a main problem with this approach: "an over-literal approach of comparing elements from games with the eight dimensions of flow does not provide desired rigor" [54]. Models are commonly inexact representations and transforming one to another may create disparities, namely showing properties not found in the process it imitates or not possessing properties owned by the process it imitates. In spite of that, these correspondences can be used as guidelines when designing the gameplay elements since they provide a general idea of what a game must contain to create flow.

<table>
<thead>
<tr>
<th>Flow Elements</th>
<th>Gameplay Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>A challenging, but tractable task to complete</td>
<td>The complete gaming experience (including social interaction during gameplay).</td>
</tr>
<tr>
<td>Full immersion in the task, no other concerns intrude</td>
<td>High motivation to play, no imperative to do otherwise; empathetic to content.</td>
</tr>
<tr>
<td>Feeling of full control</td>
<td>Familiarity/skill with controller, genre conventions, gameplay mechanics.</td>
</tr>
<tr>
<td>Complete freedom to concentrate on the task</td>
<td>Telepresence [55] and an environment dedicated to gaming.</td>
</tr>
<tr>
<td>The task has clear unambiguous goals</td>
<td>Missions, plot lines, levels; any explicit outcome of a successful play session.</td>
</tr>
<tr>
<td>Immediate feedback on actions</td>
<td>Well-timed, suitable rewards and penalties: contingencies [56].</td>
</tr>
<tr>
<td>Being less conscious of the passage of time</td>
<td>Focusing on another, temporally-independent environment.</td>
</tr>
<tr>
<td>Sense of identity lessens, but is reinforced afterward</td>
<td>Embodiment in game avatar; sense of achievement after play – e.g., ‘Hi-Score’</td>
</tr>
</tbody>
</table>

In our work, we do not provide a social interaction, since we want to have control in what game parameters have an effect on the player. Social contact could provide further effects on the mental state of the player that are out of the scope of this study. We motivated our participants in the testing phases by giving gift cards to the players that reached the highest score among all participants. Regarding feeling full control on the game, we chose a well-known and familiar genre (see Section 4.1.1 for further discussion) and provide the typical gameplay mechanics of that genre. Participants were able to fully concentrate on the task at hand, since we performed the user tests in a controlled laboratory without external stimulus. Participants also had clear unambiguous goals, since the only purpose in-game was that the player should keep on playing for as long as they could before their character died. The
game provides instant feedback depending on every action the player inputs to their character and they receive resources to continue playing the game at each specific time interval. Concerning the passage of time, our game provides a temporally-independent environment where players do not have access to the length of the gaming session. Lastly, being a FPS game, players’ sense of identity is diminished because they embody the avatar and they can only see the weapon they are carrying.

### 3.3 Adaptable Gameplay

Games in the past have been disregarding the fact that the player is a dynamic entity and therefore will evolve and adapt themself to the gaming experience. By defining, as an example, the discrete set of difficulties as easy, medium and hard, game designers aim to cover as much audience as they can. Yet as the players develop their skills concerning the game, these predetermined difficulty levels will eventually lead players out of the game designer’s scope. Other example is when the designers assume that player’s skill progression is linear and the game’s difficulty grows into a more complex state as the player progresses in the game. This approach is not appropriate since it does not take into account that some players may feel unhappy when they progress: the game counters the ‘ideal’ advancement model leading to a loss of interest in it.

In light of that, dynamic adaptable gameplay rises as the game changes its own difficulty accordingly to the player’s ongoing interactions. Since every player has a different skill growth as they play the game and based on Csikszentmihalyi’s work [11], the game’s challenge must adequate itself to the player as their skills improve in order to promote the state of flow. As mentioned before, a player will feel bored or anxious if their skills exceed or fall short of the challenge they are facing, respectively.

One of the main difficulties of this type of gameplay adaptability is to stay hidden from the player’s conscience. An affective video game must maintain an affective feedback loop controlled by the player’s physiology without the player being able to acknowledge that control and consciously control their reactions. If they do so, the video game becomes a form of biofeedback and loses its purpose. Chanel et al. [3] noted that to counter the fact that some participants totally disengaged from the game due to its difficulty, the game designer could use contextual information (e.g., current level of difficulty and the direction of the last change in it) to properly choose how the game will adapt.

Studies prove that varying the challenge level is one of the important characteristics of computer games that make them so engaging [57] and that being in a flow state may enhance learning [58, 59].

Gilleade et al. [60] proposed a challenge me gameplay: the usage of player’s arousal level as a measure to player’s engagement and use it to dynamically adjust challenge. Gilleade and Dix [4] had already used user frustration as a measure to adapt the gameplay. Rani et al. [2] used the player’s in-game anxiety and performance as measures to revamp game difficulty to keep the player as involved as possible. Comparing the two of them, they concluded that user affective feedback can be possibly
utilized to make the game more challenging, while inducing the player to perform better under lower anxiety.

A practical approach to balance the difficulty was performed by Harmat et al. [5] in a Tetris game in which the optimal difficulty (i.e., the one conceived to keep the user engaged) was based on how many complete rows the player obtained at every 30th piece. If the player managed to create three or fewer complete rows, the game’s speed would decrease one step and, if the player managed to create five or more, the speed would increase one step. This optimal condition led to the highest levels of state flow, positive affect and effortless action. Another practical example is the one of Coyle et al. [61] who adapted their game difficulty by changing the number of asteroids and/or changing their trajectory as they fall.

Nonetheless, solely using difficulty adaptation to keep user engagement is already used in various games and doesn’t bring anything new to this work. Therefore, this work not only focuses on adapting the game difficulty as the player’s mental state varies, it also creates user engagement by altering the environment and how the player interacts with it.

3.4 Classification Algorithms for Mental State Recognition


The LDA was used after the data was acquired to compute features from the signals due to the low number of samples. The classifier was trained using a cross-participant framework. In their study, Chanel et al. [3] extracted features from peripheral signals such as HR, HRV, GSR, RR and EEG features, namely alpha, beta and theta frequency bands. These features characterized physiological activity for the different conditions they tested, i.e. easy, medium and hard difficulties. Regarding the peripheral features, LDA performs better without any feature selection method. LDA coupled with an Analysis of Variance (ANOVA) feature selection on the EEG features obtained the best accuracy on finding a boundary between emotional states to generalize well across the participants.

Laine et al. [62] used several features to classify the crew members mental workload. These features included 10 seconds average of HR, HRV and time between blinks and breaths, the discrete number of blinks and breaths per 10 seconds and the average power of the alpha, beta, gamma, delta and theta EEG frequency bands. When they tried to assess the feasibility of a single model to classify mental workload across different subjects, they obtained a classification accuracy of 87%. Wilson and Russel
Lehmann et al. [64] used several classification algorithms, one of them being random forests, for recognition of Alzheimer’s disease using EEG. They obtained a sensitivity of up to 85% and a specificity of 78%. Gray et al. [65] also used random forests to assess the Alzheimer’s disease, yet they used features such as regional magnetic resonance imaging volumes, cerebrospinal fluid biomarker measures, and categorical genetic information. They obtained an accuracy of 89% between Alzheimer’s disease patients and healthy controls, and 75% between mild cognitive impairment patients and healthy controls.

Regarding Rani et al. [2], they selected, among others, GSR and EMG to distinguish the user’s anxiety state on real time and adapt the game difficulty accordingly. They were able to distinguish the state of anxiety from the state of boredom 82% of the times, from the state of engagement 76% of the times, from the state of frustration 85% of the times and from the state of anger 86% of the times.

Berta et al. [25] used SVMs to analyse EEG features, being the 36 EEG features version the best with an accuracy of 66.9% in distinguishing the user’s flow state between three different gameplay, i.e., boredom, flow and anxiety-based gaming conditions. The data sets were divided between 67% for training and validation and the remaining 33% for testing the prediction. In their work, they considered both a player-independent framework and a personalized training. They end up concluding that it is possible to train a system that can provide feedback in real time with accuracy levels 66.9% for personalized training and 50.1% for collective training.

3.5 Discussion

As presented in Section 3.1, the gaming periodicity of FPS players is an important factor so that they can process the cognitive workload present in that genre. In our study, we took this factor into account when recruiting participants. We also presented studies that address the affective state of the user, yet we use physiological measures in a different combination, as we discussed in Section 2.2.

Section 3.3 presented several ways of adapting gameplay in real-time. We adopted some of the presented design guidelines to change the game difficulty, such as increasing the number of enemies and their speed. Yet, as stated in that same section, we aim to adapt not only the difficulty, but also the way the player interacts and develops new strategies to kill their enemies. Thus, the game’s design will be focused on adapting its engine to user’s flow state, by changing the in-game difficulty and how the player interacts with the gameplay elements.

Regarding the classification algorithms, the only relevant to this work from the ones presented are the neural networks, the random forest, the regression trees and the SVM methods, since they use physiological signals as features to classify a certain factor. As we mentioned, one of the objectives of this work is to create a framework that effectively detects the user’s mental state in real time through

[63] used a neural network with the same features and the mean classification accuracies were 85%, 82%, and 86% for the baseline, low task difficulty, and high task difficulty conditions, respectively.
biofeedback measures. Thus, we modelled these four classifiers in order to choose the best for our situation, with a detailed discussion on Section 4.2.2.
4

System Overview

Contents

4.1 Game ................................................................. 27
4.2 Classification Framework ........................................ 52
4.3 Controller Framework ............................................ 66
In order to create a game which gameplay adapts to the user’s mental state, we need three different components. First, the game itself which must implement a gameplay adaptation mechanism. With this mechanism, it is possible to change the game parameters while the user is playing the game. Thus, we can change the challenge of the game by shifting the game parameters. Second, it is necessary to know the mental state of the player so that we can take a proper adaptation. We need a classification framework which function is to read the physiological signals of the player and label the output as their current mental state. Third, by knowing the current mental state of the player and taking into account the Flow Theory [1], we can decide on what to change in the game parameters so that the player is in a flow state. The controller framework is responsible for reading the current mental state of the player and varying the game parameters, depending on the previous ones. Figure 4.1 illustrates the relation between these three components and how the user is present on that relation.

Figure 4.1: User and the three core components of the solution. While the User is playing the Game, the Classification Framework is reading their biofeedback and processing it. The result is the current mental state of the player. The Controller Framework reads the current mental state of the player and decides the next game level. The Game changes its parameters accordingly to the current game level.

This chapter presents the three mentioned core components (the adaptable game, the classification and the controller frameworks) and explains the development work to obtain them.

4.1 Game

This section covers why we chose the game theme and the FPS characteristic. It also presents the base game design, how the game parameters affect the players and the different game versions we created to induce a specific mental state in the player.

4.1.1 Game Choice

The game is a FPS in a dark ambient with cartoonish zombie figures as enemies (see Figure 4.2), based on the Unity3D "Survival Shooter tutorial" [1].

The choice of a first-person camera is that it will easily lead to user immersion [66], given that one could see immersion as a precondition for flow. Note that immersion involves a loss of a sense of context, while flow describes a level of complete involvement [11]. This is possible given the removal

1 www.unity3d.com/learn/tutorials/projects/survival-shooter-tutorial
of the avatar representation and putting the player in first-person perspective, which leads the player to feel like they are acting directly upon the virtual game world \cite{67}. Thus, the player can fully identify with the game character represented only by the weapons that reach into the game environment.

The zombie genre gained popularity in recent years due to the development of themes like *The Walking Dead*\(^2\), *Resident Evil*\(^3\) and *Left 4 Dead*\(^4\). Being an endless shooter game, the product of this work uses the challenge to survive owned by the zombie genre and familiarity with the theme to create user engagement.

There are three different enemies from the tutorial: Zombunny (which have low health and high speed), Zombear (which have regular health and speed) and Hellephant (which have high health and low speed). All the enemies are pictured in Figures 4.3, 4.4 and 4.5, respectively. Also, the player only has access to a machine gun which damages one target at a time.

\(^2\)en.wikipedia.org/wiki/The_Walking_Dead_(video_game_series) 
\(^3\)en.wikipedia.org/wiki/Resident_Evil 
\(^4\)en.wikipedia.org/wiki/Left_4_Dead
4.1.2 Base Game Design

The development of the game in this phase passed through a lot of stages, mainly regarding the structure of the game world, the resources that the player has access to and how they interact with the game world and its variables.

Since the game in the end of the tutorial has an isometric camera angle, we started by changing it to a first-person camera. To do so, we imported a package from the Unity Standard Assets, called FirstPersonCharacter. This package has all the code required to control the character in a FPS game, including running and jumping, sound assets to complement and a new game object main camera. We decided to take out the functionality to jump given that, if the player jumps to the top of an object and the enemies can’t kill them, they can prolong the game indefinitely and our study is not applicable in those conditions.

As mentioned, the tutorial has an isometric view, thus this type of view led the game to have an incomplete game world, e.g. it only had two walls out of four, since it doesn’t show with that kind of camera. We had to include two more walls, a ceiling and more objects on the ground to serve as obstacles to the player. Informal user contact determined that the game world was to dark, therefore we decided to add some lamps, as can be seen in Figure 4.6, around the map.

![Figure 4.6: Game world seen from a corner of the room.](image)

Given that the player only has access to a machine gun, they can only damage one enemy at a time. This constricts the way the player interacts with the enemies, since they are only allowed to follow the same procedure. In order to allow players to create new strategies and to improve the quality of the gameplay, we decided to include a rocket launcher. A rocket launcher does damage in an Area of Effect (AOE), thus damaging any number of enemies inside a specific radius relative to the hit point. We
downloaded the model for the rocket launcher in the Unity Asset Store\(^5\). While looking for it, we tried to find the free one that had a more cartoonish look so that the game theme did not vary. The structure of the rocket launcher is similar to the machine gun’s. Both are composed by the model and an object that represents the end of the gun, \textit{i.e.} from where the ammunition is shot.

Regarding the rockets, it is also the most cartoonish looking free rocket model we could find, depicted in Figure 4.7. We developed and attached to the end of the rocket launcher a script to instantiate a rocket flying towards the direction the user is turned when they press the fire button. We also added a particle system to add smoke to the rocket so that it looks more realistic. When the rocket collides with some object, it explodes and does AOE damage to nearby enemies, as pictured in Figure 4.8. The explosion is a set of four particle systems: two create white and black smoke, one creates the red flames and the last creates debris that flies to every direction at a great speed.

![Rocket object model.](image1.png)  ![Explosion in-game.](image2.png)

Figure 4.7: Rocket object model.  Figure 4.8: Explosion in-game.

After creating the model, we wrote a script and attached it to the player so that they can switch between both weapons when they press a specific key. When a weapon is not being used by the player, the object which represents it is set to inactive. Thus, every script it has attached is not run and its model is not rendered.

Given that the perspective of the weapon is purposely not perpendicular to the screen, players are not able to intuitively know where they are shooting at. In order to help the user shooting at far away enemies and reducing the cognitive workload, we implemented an aiming system, which is a script attached to the player object that changes both the positioning of the gun and the first-person camera when they press a button. This way, players can correctly aim at enemies that are at a considerable distance. We also implemented an aim assist script. Its function is to drastically reduce the mouse sensibility when the user hovers an enemy, easeing the aim. Lastly, we added a red crosshair in the center of the screen, which allows the player to have a reference of the hit point without needing to aim.

In order to prolong the game, the user must have resources they can use to fight the enemies. We decided to include a pick-up system that generates resources through all the map. The pick-ups

\(^5\)www.assetstore.unity3d.com/en/
have three different types and the models were downloaded from a free asset available in the Unity Asset Store. When the pick-ups are available for the player to pick them up, they are rotating around an axis perpendicular to the ground and have a color associated for each type. The health pick-up is red (Figure 4.9), the ammunition for the machine gun is yellow (Figure 4.10) and the rocket launcher ammunition is blue (Figure 4.11).

Figure 4.9: Health pick-up in-game.  
Figure 4.10: Machine gun ammo pick-up in-game.  
Figure 4.11: Rocket launcher ammo pick-up in-game.

When the player gets an health pick-up, they receive 25% of their maximum health. If they have 75% or more health, the health is capped at 100%. The machine gun ammo pick-up gives the player 50 bullets and the total number they can carry is 200. Finally, the rocket launcher ammo provides an additional 5 rockets and the player cannot carry more than 15.

We randomly chose the spawn points set for the pick-ups along the map so that the player can create their own routes to pick them up. The pick-ups spawn at a given time in a randomly chosen point from the whole set. If the chosen point has already a pick-up there, it does not spawn another one in the same point. The pick-up that spawns in a chosen point is one of the three previously mentioned and the probability for each type spawning can be set by the developer.

Although the FirstPersonCharacter already allowed the player to run, if the player could run indefinitely the game would not be realistic. The stamina adds a level of realism and players will only be able to sprint for a limited period of time. This way the stamina concept is used as a way to limit players ability to do repeated actions and to provide an element of strategy. The player restores their stamina two times faster if they are not moving than when they are walking. This also allows the player to explore new strategies.

The Graphical User Interface (GUI) at the end of the tutorial only showed the score and total life of the player. We complemented it by showing the number of bullets and rocket ammunition the player had against the total number of each ammunition they can carry. We also implemented a stamina bar so that the player knows in each moment how much they can run before it ends. The new GUI is pictured in Figure 4.12.

In order to have a way to analyze player progression along a game session, we created an object
called Output Manager which has two main functions. While playing, at every \( n \) seconds defined by the developer, the output manager writes all the game variables in a text file so that we can study player evolution. Moreover, after the player dies and the final score is presented to them, the output manager writes in another text file the final state of the game.

One problem with the tutorial is the output sound mixing. Every enemy produces a sound when it dies. If the player kills more than one enemy of the same type, the sound output is cumulative rather than only playing a death sound to represent all the dying enemies. Thus, instead of outputting only one meek sound, the game produces multiple copies of the same sound and the level of decibels is so high that it physically hurts the player. To correct this, we created an object called Sound Manager. Its job is to control the number of death sounds the enemies produce. We implemented it in a way that, if a death sound from among the three possible was played in the last 0.2 seconds, the same death sound could not be played again. This way, when a lot of enemies of the same type are killed at the same time, the player only hears one time the death sound of that type of enemy.

To finish the base game, we created a sandbox version of the game. This version is used to allow players to develop their initial skills, as mentioned in Section 2.1 and only spawns enemies when a specific key is pressed and the pick-ups spawn regularly.

### 4.1.3 Adaptable Game Parameters

The most relevant parameters regarding difficulty in our game are the ones of the enemies. Taking into account existing studies about flow during gameplay [5, 13] as well as informal feedback from users, we decided to address specially the speed, health and spawn time of the enemies. Figures 4.13, 4.14 and 4.15 show sketches based on player’s intuition regarding how users felt while play testing differences in the game parameters values of enemies’ speed, health and spawn time stats. The plots are not validated. Although it would be interesting to validate these plots, the validation is out of the scope of our main contributions.
Figure 4.13: Sketch of the mental states induced by the game as a function of the character parameters health and speed. If all characters are too slow and easy to kill, the game is too easy and the player becomes bored. As the value of the two parameters increases, the user becomes increasingly engaged and eventually anxious and frustrated. Finally, if the characters are too slow, but have excess health, they are easy to kill but doing so takes too much time, eventually inducing apathy in the player.

Figure 4.14: Sketch of the mental states induced by the game as a function of the enemy parameters health and spawn time. A balance between the values of the two parameters leads the user to engagement. If enemies have a short spawn time, the player becomes anxious, but, if enemies also have excess health, the player becomes frustrated. Finally, if the spawn time is high, players become bored, except if they also have excess health, which leads players to an apathetic state.
Figure 4.15: Sketch of the mental states induced by the game as a function of the enemy parameters speed and spawn time. Testers said that, independently of the speed, for high values of spawn time, they would be bored and, in the limit, apathetic. As the value of the speed parameters increases and the spawn time decreases, the user becomes decreasingly engaged and eventually anxious and frustrated.

Other parameters such as player’s running speed, shooting probability and pick-ups spawn probabilities are also noteworthy. Regarding the player’s running speed, users claimed that their engagement was directly proportional to it, i.e. users felt that the game had more action. The shooting probability is the chance that, if the player pulls the trigger, the bullet is fired. Accordingly to users, their frustration levels increased as the shooting probability decreased, since they were not able to kill the enemies in spite of having the means to do so. Lastly, the pick-up spawn probability also had an important factor in the user. Players preferred a higher rocket launcher ammunition probability compared to the machine gun ammunition and health probabilities, claiming that it led to a higher engagement. We expect this to be a result from the type of damage that the rocket produces when it collides with an object and the amount of damage it does to enemies.

With these parameters and the feedback from users, it is possible to create versions of the game that induce a certain mental state in the user. A key point in our work is that our contribution must be able to distinguish the mental state of the user. Therefore, we need to have data of the different mental states. In order to investigate the assessment of the mental state of the players as they interact with the game, we created four different versions thereof by varying some of the game parameters - the speed, health and spawn time of the enemies.

4.1.4 Inducement Versions Design

Each version was fine-tuned to induce in the player a specific mental state, such as anxiety, boredom, engagement and frustration. The choice of which mental states to induce was based on the works of
Players will also be able to play a sandbox version of the game and develop their initial skills. This takes away the need to create a game version focusing on apathy, since participants will not have skills too low for the task.

The versions and their variations are presented in Table 4.1. In addition to the game parameters values of enemies’ speed, health and spawn time stats, we decided to alter other variables in versions B and D. In version B, the player’s speed is reduced, the pick-ups’ spawn time is enlarged, the ammunition spawn probability is increased and the health pick-up probability is reduced. Such settings make the game less engaging, since the player cannot run fast, which is expected to induce boredom in the player, and avoid apathy, since the player has more ammunition to fight back. In version D, the player’s shooting probability is reduced. The variation makes the game too hard, which is expected to induce frustration in the player. The differences with specific values between the versions can be seen in Table 4.2.

**Table 4.1:** Version name, the mental state it is intended to induce and the general characteristics of each game version.

<table>
<thead>
<tr>
<th>Version</th>
<th>Mental State</th>
<th>Parameters Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Anxiety</td>
<td>Low enemy health</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High enemy speed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low enemy spawn time</td>
</tr>
<tr>
<td>B</td>
<td>Boredom</td>
<td>High enemy health</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low enemy speed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High enemy spawn time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low player speed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High pick-up spawn time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High ammunition pick-up probability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low health pick-up probability</td>
</tr>
<tr>
<td>C</td>
<td>Engagement</td>
<td>Medium enemy health</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Medium enemy speed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Medium enemy spawn time</td>
</tr>
<tr>
<td>D</td>
<td>Frustration</td>
<td>Low enemy health</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High enemy speed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low enemy spawn time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low shooting probability</td>
</tr>
</tbody>
</table>

**4.1.5 User Studies**

With the inducement versions created, we run a testing phase to validate them. In this testing phase, we also decided to record the physiological signals from the users while they played the different versions of the game, in case we validated the inducement versions and we decided to use these biofeedback data to model a classifier. This section explains all the procedures and presents the results we obtained regarding the inducement versions.
<table>
<thead>
<tr>
<th>Variable</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zombunny Spawn Time</td>
<td>2</td>
<td>8</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Zombunny Speed</td>
<td>7</td>
<td>2.5</td>
<td>5.5</td>
<td>7</td>
</tr>
<tr>
<td>Zombunny Health</td>
<td>40</td>
<td>100</td>
<td>60</td>
<td>40</td>
</tr>
<tr>
<td>Zombear Spawn Time</td>
<td>3</td>
<td>10</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Zombear Speed</td>
<td>6</td>
<td>2</td>
<td>4.5</td>
<td>6</td>
</tr>
<tr>
<td>Zombear Health</td>
<td>80</td>
<td>200</td>
<td>100</td>
<td>80</td>
</tr>
<tr>
<td>Hellephant Spawn Time</td>
<td>8</td>
<td>20</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>Hellephant Speed</td>
<td>3</td>
<td>1.5</td>
<td>2.5</td>
<td>3</td>
</tr>
<tr>
<td>Hellephant Health</td>
<td>240</td>
<td>500</td>
<td>340</td>
<td>240</td>
</tr>
<tr>
<td>Pick-Ups Spawn Time</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Gun Ammo Probability</td>
<td>35%</td>
<td>40%</td>
<td>35%</td>
<td>35%</td>
</tr>
<tr>
<td>Rocket Ammo Probability</td>
<td>35%</td>
<td>40%</td>
<td>35%</td>
<td>35%</td>
</tr>
<tr>
<td>Health Probability</td>
<td>30%</td>
<td>20%</td>
<td>30%</td>
<td>30%</td>
</tr>
<tr>
<td>Player Run Speed</td>
<td>10</td>
<td>7</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Shooting Probability</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.9</td>
</tr>
</tbody>
</table>

### 4.1.5.1 Participants

Subjects were recruited through standard procedures including direct contact and through word of mouth. Subjects included anyone interested in participating if they were at least 18 years old. Each participant was asked to sign a consent form. There were no potential risks and no anticipated benefits to individual participants.

A total of 31 participants performed our tests. Due to technical problems, one test was discarded. All tests occurred between 08:30h and 20:00h. A total of 30 fully completed tests were used for the analysis. The participants (22 males, 8 females) were ranged in age from 19 to 27 ($M = 21.87, SD = 1.57$).

Only six participants reported no video game-playing time. The other participants play at least once a day (13.3%), at least once a week (33.3%) or at least once a month (33.3%). Regarding FPS games, nineteen participants play that type at least once a month. We performed a mixed ANOVA to verify if the difference in cognitive flexibility [46] between participants that played frequently FPS and players who did not play that genre frequently. Results showed that gaming periodicity and gender had no significant effects on results.

### 4.1.5.2 User Evaluation

This section introduces the two questionnaires used along the testing sessions and a way to measure flow during the gameplay. We use the PANAS [69] to perceive if the positive and negative emotions changed significantly between the gaming sessions. This is important because the player must have higher positive and low negative affects to reach flow, otherwise they are not enjoying the game. Concerning flow measurement, there are various studies on scales to measure it, being it in e-learning.
games [19], in action games [48] and in exercise games [70]. We are using an adapted version of the GEQ [48] to assess the player’s flow components after they played a version of the game. We are not using the full GEQ since many of its components are not relevant to our study. From the second questionnaire comes the Flow Degree, a measure we created to classify the flow in a discrete number within an interval.

**PANAS**

Watson *et al.* [69] developed the PANAS scale with the objective to assess one’s self-perceived well-being and affectivity. The original scale consists of twenty items that aim to evaluate positive and negative affect on a Likert scale ranging from “Very slightly or not at all” (1) to “Extremely” (5). Of the twenty items, ten belong to the positive affect component, *e.g.*, enthusiasm, inspiration, interest, and the other ten to the negative affect component, *e.g.*, irritation, fear, nervousness [69].

The Portuguese version of PANAS [71] includes eleven items to evaluate positive affect and eleven items to evaluate the negative affect, having one more item in each component compared to the original scale. Simões [71], found a Cronbach alpha of 0.82 for the subscale of positive affect and 0.85 for the subscale of negative affect.

Our study uses the Portuguese version of the PANAS with only the original twenty items developed by Watson *et al.* [69], as illustrated in Figure 4.16. In this testing phase we use two PANAS scales. The user fills one of them is filled before they start the gaming sessions in order to have a reference of how the player felt before starting the tests. The second scale is filled after the gaming sessions.

**Adapted GEQ**

The questionnaire contains ten questions on a 10-point Likert scale. It consists of the following items in English:

- I was getting tired;
- I felt like I could not stop playing;
- I played without thinking how to play;
- I lost track of time;
- I got wound up;
- I felt anxious;
- I felt bored;
• I felt frustrated;
• I liked to play the game; and
• I would like to play the game again.

The first five items are taken from the GEQ [48], with four of them regarding elements of flow and one concerning presence. The first element in the original GEQ version was “I can’t tell I’m getting tired”, but, given a mistake in the translation from English to Portuguese, it was used as “I’m getting tired”. There is no difference since one is the negation of the other. It relates to the flow dimension concerning the loss of self-consciousness, the second question regards the autotelic dimension, the third one has to do with the merging of action and awareness and the fourth point addresses the balance between the challenge of the task and the individual’s skills. The element concerning presence can be representative of the flow dimension regarding the time transformation in which the sense of time is crooked.

The three next questions consist of the self-assessment of anxiety, boredom and frustration since these three mental states are the complements of the flow state. In case that people are in either one of those states, they are not into flow.

The last two items have no purpose for our study and are only used to mislead the user. This helps preventing the user from trying to guess what was special about the version they played and not focus on the task at hand. The final form with the order of the items randomized is illustrated in Figure 4.17.
Agora pedimos que responda a umas perguntas sobre a versão do videogame que acabou de jogar. Leia cada afirmação e marque a resposta adequada de 1 a 10, sendo que 1 representa Discordo totalmente e 10 representa Concordo totalmente. Indique em que medida concorda com as seguintes frases:

<table>
<thead>
<tr>
<th>Frase</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eu sinto que estava a ficar cansado(a) enquanto jogava.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu gostei de jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu sinto que não ia conseguir deixar de jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu gostaria de voltar a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu estava a jogar sem pensar em como o fazia.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu senti-me aborrecido(a) enquanto estava a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu perdi noção do tempo enquanto estava a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu senti-me frustrado(a) enquanto estava a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu senti-me preocupado(a) enquanto estava a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eu senti-me ansioso(a) enquanto estava a jogar.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure 4.17: Adapted GEQ scale from the user form.*
Flow Degree Scale

We can assess if the participant felt anxious, bored or frustrated because there is an explicit item that addresses those states. Yet, we cannot do the same for flow. Since we cannot say that a user was in flow just by addressing individual flow elements, we decided to create the Flow Degree Scale.

This is a scale that uses the items from the Adapted GEQ questionnaire and helps to distinguish when the user felt engaged from anxious, bored and frustrated. The Flow Degree Scale gives a discrete Flow Degree number and it uses the first five items. This Flow Degree number is calculated by the sum of each points from the items "I felt like I could not stop playing", "I played without thinking how to play" and "I lost track of time" minus the points from the items "I was getting tired" and "I got wound up". The values vary between −17 and 28. We cannot compare the values from 1 to 10 from the anxiety, boredom and frustration items to the Flow Degree because of the different value intervals. With this in mind, when compared to the other mental states, Flow Degree was normalized using the min-max normalization.

This scale will be later used to label the recorded physiological data accordingly to the highest value among the anxiety, boredom, frustration items and the normalized Flow Degree.

4.1.5.3 Apparatus

In Section 2.2, several physiological measures were discussed in order to find the most effective ones regarding the flow state detection. We used a BiTalino⁶ to record the Blood Volume Pulse (BVP) with a photoplethysmography sensor, the electricity on the skin of the hand with two electrodes and the electricity on the forehead with three electrodes. The first allows us to know the instantaneous HR, the second the GSR and the third the brainwaves (alpha, beta and theta bands). All these items can be obtained in the (r)evolution Plugged Kit BT⁷ with an addition of a PulseSensor⁸. The BiTalino was placed on a table behind the users and the sensors connected to it in each experimental condition and during the baseline measurement, as seen in Figure 4.18. While they were playing, participants had an earplug attached to one of their ears, two electrodes on their left hand with a band to help keeping them in place and two electrodes on their forehead (positions FP1 and FP2 in the 10–20 system [72]), one electrode on the left side of their neck (its function is to serve as "ground" for the difference in both hemispheres). The second band was only used for participants who had long hair covering their forehead. These sensors are displayed in Figures 4.19, 4.20 and 4.21. Users were seated on a chair with the computer in front of them on top of a table, as seen in Figure 4.22.

We needed two computers for this testing phase. Data was recorded at 100 Hz with OpenSignals (r)evolution Mac OS X (v.2017)⁹ software. Both computers were capable of processing and running the programs without any delay. Players interacted with the game through mousepad and headphones.

---

⁶www.bitalino.com/en/hardware
⁷bitalino.com/en/plugged-kit-bt
⁸store.plux.info/bitalino-sensors/42-pulsesensor.html
⁹www.bitalino.com/en/software
also carefully chose seven images depicted in Figures 4.23 and 4.24 to lead to a relaxing state. Other materials used for the experiences were two ribbons, neurodiagnostic electrode paste, alcohol and cloth.

![Main board of the (r)evolution Plugged Kit from Bitalino with three sensor cables connected to it.](image1)

![Forehead sensors.](image2)

![Earplug and neck sensors.](image3)

![Hand sensors.](image4)

### 4.1.5.4 Procedure

Tests were performed in room 0.09 of Pavilhão de Informática II in campus Alameda of Instituto Superior Técnico (IST). Temperature was kept between 21°C and 23°C in order to prevent the electrode gel from melting and displacing the sensors.

Participants were required to play each version of the game for ten minutes, adding to a total of forty
If we include the expected time to set up the sensors (three minutes), the sandbox playtime (five minutes), the time to fill the forms (eight minutes) and all the resting phases (five minutes for baseline recording and three rest phases of three minutes each), each test took around seventy minutes.

Upon arrival, the assistant explained the participants the purpose of the study and what they would be doing. The assistant then placed the sensors on the participant, whom was asked to fill a consent form and a questionnaire regarding their demographic data, gaming experience and emotional state with PANAS. The latter form was used for the rest of this experience to fill the adapted GEQ for each version. Note that the order of the items was randomized between the four versions. After filling it, physiological sensors were placed on the participant.

Before playing any version of the game, users were asked to rest to record their resting physiological data. While the players were resting, they were looking at a set of images, as depicted in Figure 4.22. After that, users were given a sandbox version of the game to try the sensitivity and in-game interactions. The tester could play as much time as they wanted in order to develop the minimum skills to play the game in the different versions. Each gaming session was ten minutes long and the order by which each player interacted with the four versions of the game was randomized. The following procedure was used for each version the participant played:

1. The assistant verified if every sensor was correctly placed;
2. The assistant asked the tester if they was comfortable and ready to play;
3. The assistant started the game for the tester;
4. If the player died, the game automatically restarted;
5. The player played for ten minutes, as seen in Figure 4.25, and after it the game automatically closed;
6. The assistant asked the tester to fill the form addressing the version the latter played;
7. The player rested for three minutes looking at a picture (see Figure 4.23) in order to return to a neutral mental state; and
Figure 4.24: Set of chosen pictures for the control test. The chosen images are part of a research program from the American National Institute of Mental Health called Center for the Study of Emotion and Attention. Its function is to create and distribute affective ratings for stimuli in studies of emotion and attention. These images are part of the IAPS [73] and they are only allowed to be used in the academic field. Studies regarding emotions [74, 75, 76] found reliable changes in autonomic and somatic psychophysiology as well as in brain activity (e.g., EEG, fMRI) when viewing pictures from the IAPS. Lang [74] found that a relaxing feeling can be evoked by pictures of flowers. The images were proved to lead to high values of valence and low values of arousal, which correspond to relaxing stimuli. The caption of each picture is its identifier in the IAPS.
8. After three minutes, the assistant repeated this procedure to the next version, if there was any other version to play.

![Tester playing one of the versions of the game in the laboratory.](image)

Figure 4.25: Tester playing one of the versions of the game in the laboratory.

The decision regarding the game closing automatically after ten minutes seeks to avoid breaking immersion (by talking to the subject) before ten minutes of game have passed.

After the four sessions were played, the assistant removed the sensors from the tester and cleaned them with alcohol to remove the electrode gel. When the cleaning was done, the assistant asked them to fill another form with a PANAS scale. Free comments were also invited.

Testers received a compensation based in candies. There was also a contest to see which player achieved the highest score across all gaming sessions. The winner received a gift card worth 20EUR.

### 4.1.6 Analyzing Data

Regarding data from questionnaires, we did not need further processing for analysis. Subsections 4.1.6.1 and 4.1.6.2 address all the useful information that we could extract from them, namely how the players’ positive and negative affects varied and if the design guidelines were well applied to induce the desired mental states.

We performed Shapiro-Wilk normality tests with all the data to determine which follow-up tests to use. Regarding PANAS, we analyzed significant differences in positive and negative affects between before and after the gaming sessions to check if users liked the game and if the gaming length was not too long. We also tried to see if there were a significant difference between the self-perceived mental states from the users and the inducement version the player was playing, allowing us to validate our design guidelines. These results are presented in the next subsections along with the hypotheses that we aim to test.

#### 4.1.6.1 PANAS

Results from the PANAS questionnaire allow checking if, in general, the users liked to play the game and if the length of the testing session was too long. This leads us to investigate the following hypothesis:
**H1:** Users have a lower positive affect before the gaming sessions than after it.

This hypothesis states that playing the game induced a significant difference in the positive affect of the user before and after the game testing, which may suggest that the player enjoyed playing the game. Since we are doing four sessions of ten minutes each, fatigue and frustration may rise in the player’s affective state. We can investigate another hypothesis:

**H2:** Users have a lower negative affect before the gaming sessions than after it.

This hypothesis states that the game testing length induced a significant difference in the negative affect of the user before and after the game testing, which may suggest that the length is too long and it leads players to negative emotions.

**H1: Users have a lower positive affect before the gaming sessions than after it**

Although the Shapiro-Wilk test was not significant for data before the gaming session ($D(30) = 0.977, p > 0.05$), it was significant for data after the gaming session ($D(30) = 0.913, p < 0.05$). The Wilcoxon test’s $z$-score is $-2.479$ and this value is significant at $p = 0.013$. Therefore, we can conclude that the degree of positive emotions of the testers before the gaming sessions ($Mdn = 27.5$) was significantly less than after the gaming sessions ($Mdn = 33$), $T = 103.00, p < 0.05, r = -0.32$. Therefore, Hypothesis 1 is accepted. The boxplots in Figure 4.26 show our results and we observe that both datasets have a similar Interquartile Range (IQR).

![Figure 4.26: Boxplots of positive affect for Before and After the gaming sessions.](image-url)
**H2: Users have a lower negative affect before the gaming sessions than after it**

The Shapiro-Wilk test was significant for data before the gaming session \( (D(30) = 0.751, p < 0.001) \) and for data after the gaming session \( (D(30) = 0.846, p < 0.05) \). The Wilcoxon test's \( z \)-score is \(-1.717\) and this value is not significant at \( p = 0.086 \). Therefore, it appeared that the degree of negative emotions of the testers before the gaming sessions \( (Mdn = 11) \) was not significantly different compared to after the gaming sessions \( (Mdn = 12.5) \), \( T = 138.50, ns, r = -0.22 \). Therefore, the test is inconclusive for Hypothesis 2. Yet, by observing the median values from both times in Figure 4.27, it is possible to observe that after the game session the negative feelings were higher than before it. Also, negative affect after the gaming session has a larger IQR compared to before it.

![Boxplots of negative affect for Before and After the gaming sessions.](image)

**Figure 4.27:** Boxplots of negative affect for Before and After the gaming sessions.

Regarding **H1**, since both conditions have a similar IQR, results suggest that they have a similar variation. The difference between the minimum and maximum values (excluding outliers) for both conditions shows that, although there are three outliers, people feel their emotions in a different way and positive affect after the gaming session has generally higher values compared to before it. As we accepted **H1**, we conclude that the participants liked the theme and to play the game. Regarding the negative affect, although not at a significant level, players showed a little more negative affect after the gaming sessions. This can also be verified by the larger IQR after the gaming sessions, which indicates that the value variation was also larger compared to before the gaming sessions. It was expected since players needed
to be playing the game for, at least, forty minutes and we induced negative emotional states (anxiety, boredom and frustration).

4.1.6.2 Adapted GEQ

The Adapted GEQ we created allows us to check two things: which was the strongest mental state the user felt playing a certain version and validate if the design guidelines created for each version were appropriated. Knowing which was the strongest mental state is important to model the classifier as you will see in Section 4.2.2. This section addresses the validation of the design guidelines for each version and if they successfully induced the respective mental states in the participants based on the respective item, i.e. we checked if the version A led the player to an anxiety state based on their responses on the "I felt anxious" item. This leads us to create four more hypothesis:

H3: Version A leads to a higher anxiety state compared to B, C and D versions.

H4: Version B leads to a higher boredom state compared to A, C and D versions.

H5: Version C leads to higher Flow Degree values compared to A, B and D versions.

H6: Version D leads to a higher frustrated state compared to A, B and C versions.

H3: Version A leads to a higher anxiety state compared to B, C and D version

The Shapiro-Wilk test was significant for the data generated after playing versions A \( (D(30) = 0.920, p < 0.05) \), B \( (D(30) = 0.901, p < 0.05) \), C \( (D(30) = 0.922, p < 0.05) \) and D \( (D(30) = 0.925, p < 0.05) \). The version the tester played significantly affected how they answered the item "I felt anxious", \( \chi^2(3) = 19.939, p < 0.001 \). Wilcoxon tests were used to follow-up this finding using version A as the control one. We applied a Bonferroni correction and so all effects are reported at a 0.0167 level of significance.

It appeared that playing version A would make the user significantly more anxious compared to version B \( (T = 66.50, r = -0.360) \) and less anxious compared to version D \( (T = 82.50, r = -0.311) \). However, how much anxious the player would be after playing version C \( (T = 136.00, ns, r = -0.133) \) compared to version A was not significantly different. We can conclude that playing version B led to a less anxious feeling and version D to a more anxious feeling compared to version A, and this effect was medium in size for both versions. Nevertheless, version C did not produce any substantial difference of anxiety relative to the control programme. Therefore, this test is inconclusive for this hypothesis.

In Figure 4.28, version D has the highest median \( (Mdn = 7) \), compared to the ones of versions A \( (Mdn = 4) \), B \( (Mdn = 4) \) and C \( (Mdn = 4.5) \). This suggests that, while not significantly, users were more anxious playing version D compared to any other. Version A and version D have a large and a small IQR, respectively. Values for versions B and C have similar IQR, which suggests that they have a similar variation.
H4: Version B leads to a higher boredom state compared to A, C and D versions

The Shapiro-Wilk test was significant for the data generated after playing versions A \((D(30) = 0.880, p < 0.05)\), B \((D(30) = 0.919, p < 0.05)\), C \((D(30) = 0.825, p < 0.001)\) and D \((D(30) = 0.911, p < 0.05)\). We conclude that the version the tester played didn’t significantly affect how they answered the item “I felt bored”, \(\chi^2(3) = 3.652, p > 0.05\). We conclude that this test is inconclusive for this hypothesis.

Figure 4.29 pictures the boxplots of the medians of the answers for this item across different versions. Version B has the highest median \((Mdn = 4)\), compared to versions A \((Mdn = 3)\), C \((Mdn = 2)\) and D \((Mdn = 3)\). This suggests that, while it is not significant, users were more bored playing version B compared to any other. It is also noteworthy that versions A and B have a similar and larger IQR compared to versions C and D, which also have a similar IQR.

H5: Version C leads to higher Flow Degree values compared to A, B and D versions

The Shapiro-Wilk test was non significant for versions A \((D(30) = 0.969, p > 0.05)\), B \((D(30) = 0.955, p > 0.05)\), C \((D(30) = 0.975, p > 0.05)\) and D \((D(30) = 0.984, p > 0.05)\). We conducted a one-way repeated measures ANOVA to compare the effect of the gaming version on value of Flow Degree in versions A, B, C and D. Mauchly's Test indicated that the assumption of sphericity has been violated, \(\chi^2(5) = 19.407, p = .002\). There was a non significant effect of which version the participant
played, Wilks' Lambda = 0.758, $F(3, 27) = 2.871, p = 0.055$. Greenhouse-Geisser correction determined that mean Flow Degree value did not differed statistically significantly between the gaming versions $F(2.020, 58.579) = 1.814, p = 0.172$. Although the test for this hypothesis is inconclusive, we can verify in Figure 4.30 that version C had the highest median ($Mdn = 6.5$), compared to the ones of versions A ($Mdn = 6$), B ($Mdn = 6.2$) and D ($Mdn = 5.9$). Also, all versions have values with similar IQR.

**H6: Version D leads to a higher frustrated state compared to A, B and C versions**

Shapiro-Wilk test was significant for the data generated after versions A ($D(30) = 0.930, p = 0.05$), B ($D(30) = 0.837, p < 0.05$) and D ($D(30) = 0.930, p > 0.05$). Version C data was non significant ($D(30) = 0.937, p > 0.05$). We can conclude that the version the tester played significantly affected how they answered the item "I felt frustrated", $\chi^2(3) = 46.460, p < 0.001$. Wilcoxon tests were used to follow-up this finding using version D as the control one. We applied a Bonferroni correction and so all effects are reported at a 0.0167 level of significance.

It appeared that playing version D would frustrate the user to a significantly higher compared to version A ($T = 12.50, r = -0.394$), to version B ($T = 3.50, r = -0.588$) and to version C ($T = 36.00, r = -0.425$). We can conclude that playing versions A and C led to a less frustrated feeling compared to version D, and this effect was medium in size for both versions, and that playing version B led to a less frustrated feeling compared to the control one with a effect large in size. Thus, this hypothesis is
Figure 4.30: Boxplots of the medians for the normalized Flow Degree values.

Figure 4.31 pictures the boxplots of the medians of the answers for this item across different versions. As expected, version D has the highest median \((Mdn = 7)\), compared to the ones of versions A \((Mdn = 5.5)\), B \((Mdn = 2.5)\) and C \((Mdn = 6)\). Versions A and B have similar IQR. In contrast, version C and version D have large and small IQR.

Tests for H3 were inconclusive, but we were able to find out that it let users significantly more anxious than version B and less anxious than version D. It is expected that the user felt more anxious in version D than in version A, since frustration is the extreme of anxiety. Thus, users usually feel greater anxiety when they are frustrated rather than when they are simply anxious. Regarding distinguishing engagement from anxiety, we were not able to assess if the design guidelines provided a significant difference. Note that the median for the item "I felt anxious" was higher in version C than version A, yet the difference was minimal. Also, with a small IQR, the variation of values in version D was much smaller compared to the other versions. We think that it happened because the enemies had more health points that they should have and it led users to feel more anxious because it took too long to kill one enemy. With this in mind, results suggest that our design guidelines for this version were accurate to induce anxiety.

Although H4 did not provide significant results, the median of the item "I felt bored" was higher on that version compared to any other, although the IQR was similar to the one of version A. This means that, compared to the other versions, version B induced a higher boredom state in the players than
other versions. Therefore, results suggest that we were able to successfully induce boredom on the participants with our guidelines.

**H5** analysis shows that there was no significant difference between version. In spite of not being able to verify that our guidelines were statistically significant, we see that the Flow Degree median was higher for version C and that all versions showed values above five in a ten-point Likert scale, meaning that, in general, people felt engagement in every version they played. Also, the IQR was similar for every version, so we can conclude that they all have similar variations. Thus, results suggest that our guidelines for version C were able to induce engagement.

Lastly, the guidelines used for version D were a success, since there were significant differences between the values in the answers for the "I felt frustrated" item between versions. The highest median and the smallest IQR for this version also show those results. We conclude that our guidelines for version D were able to induce frustration.

### 4.1.7 Discussion

This section explained the design process of the game and its adaptable parameters. It continued by illustrating how the variation of those parameters impacted the mental state of the users. Lastly, in order to create the framework to process biological signals and detect the mental state of the player, we needed to understand how the signals varied among mental states. Thus, we created and validated
different game versions to induce specific states on the player.

The first testing phase provided important results for the development of the solution. PANAS results suggest that, not only players significantly enjoyed the theme and gameplay, but they were not significantly negative after playing so much time. These results show that the theme and gameplay were adequate and that we should extend them while developing the adaptability of the game. Also, testing time of the second testing phase should be less than forty minutes, since that was the total game time players went through in the first testing phase. Regarding the guidelines we used for the inducement versions, results suggest that versions A, B and C induce anxiety, boredom and engagement, respectively. Moreover, version D significantly induces frustration. Therefore, we can use physiological data collected in this testing phase to model our classifier. We also conclude that these guidelines can be used to increase difficulty in the final solution, yet they must be adjusted to provide significant results. The next section covers how the processed data obtained from the testing phase were a core piece for the creating of the classification framework.

4.2 Classification Framework

The main function of the classification framework is to detect the mental state of the user. The detection of the mental state is carried out by a classifier. As mentioned in the previous section, all data obtained from performing the previous tests to the users is the feedstock for the classifier. We need to model and train the classifier with these data so that the final solution can distinguish the mental states we aim to address among themselves. This section addresses the validation of the physiological measures to be used when distinguishing the mental state of the user and the modeling of the classifier. We also present the final architecture of this main component.

4.2.1 Choosing the Physiological Measures

As mentioned in Section 4.1.5.3, the physiological signals we aim to measure are the brainwaves (alpha, beta and theta bands), HR and GSR of the participants. In order to have them, firstly we need to process data. We used a set of a photoplethysmography sensor and five electricity sensors to collect the biofeedback from the user (see Figures 4.19, 4.20, 4.21 and 4.22). Bitalino reads the analog signals from the participants and transforms them in digital signals. These signals are not meaningful per se and should be converted to the physiological measures presented in Section 2.2. Thus, we converted them to BVP, EDA and EEG values. We used the BioSPPy toolbox to compute the instantaneous HR and GSR, and EEG alpha, beta and theta bands.

After we obtained the data, we can start studying which measures are best, if any, to distinguish the mental state of the user. We investigate five new hypotheses, one for each physiological measure of this
study. These hypotheses state that the different versions of the game induced significant differences in signals measured in a specific physiological measure, which may suggest that signal may be relevant to predict changes in the mental state of the player.

**H7**: Values for the alpha band are different between game versions.
**H8**: Values for the beta band are different between game versions.
**H9**: Values for the theta band are different between game versions.
**H10**: Values for the HR are different between game versions.
**H11**: Values for the GSR are different between game versions.

Given that physiological data is not normalized, we decided to use the median of the records as a reference to compare the different versions because the effect of outliers in the median is very small or not existent. Therefore, we our processing is independent of outliers. The values we used for comparison are the medians of the result of subtracting the values of the records for each version and the median value of the control record. We can then create a classifier that does not depend on a person, but rather addresses the differences that people feel between different mental states. We assume that these differences are universal.

Due to technical limitations of the acquisition hardware, the skin resistance of many participants was outside the range that the sensor could measure. Such limitations caused around 19% of missing values in the skin resistance data. We completed these missing values with expectation maximization (see [77, 78] for further discussion). It overcomes some of the limitations of other techniques, such as mean substitution or regression substitution. The two latter alternative techniques generate biased estimates and, specifically, underestimate the standard errors.

**H7: Values for the alpha band are different between game versions**

The Shapiro-Wilk test was significant for the data generated after playing versions A \( (D(29) = 0.858, p < 0.05) \), B \( (D(29) = 0.837, p < 0.05) \), C \( (D(29) = 0.865, p < 0.05) \) and D \( (D(29) = 0.861, p < 0.05) \). There was no significant difference between the alpha band values corresponding to the different versions of the game, \( \chi^2(3) = 1.345, p = 0.719 \). Then, we can refute hypothesis 7.

**H8: Values for the beta band are different between game versions.**

The Shapiro-Wilk test was significant for the data generated after playing versions A \( (D(29) = 0.881, p < 0.05) \), B \( (D(29) = 0.891, p < 0.05) \) and C \( (D(29) = 0.917, p < 0.05) \). Version D was not significant for this test \( (D(29) = 0.937, p = 0.086) \). There was a significant difference between the beta band values corresponding to the different versions of the game, \( \chi^2(3) = 10.697, p = 0.013 \). We applied Wilcoxon Signed Rank tests between all the combinations of the versions.
It looks like we can only significantly differentiate the beta band waves between the pairs Engagement-Boredom \((T = 76.00, r = -0.4018)\) and Frustration-Boredom \((T = 66.00, r = -0.4281)\), out of the six possible pairs. This suggests that, although the test for this hypothesis is inconclusive, there is a significant difference between the beta band among gaming versions.

**H9: Values for the theta band are different between game versions.**

The Shapiro-Wilk test was significant for the data generated after playing versions A \((D(29) = 0.849, p < 0.05)\), B \((D(29) = 0.907, p < 0.05)\), C \((D(29) = 0.871, p < 0.05)\) and D \((D(29) = 0.866, p < 0.05)\). There was no significant difference between the theta band values corresponding to the different versions of the game, \(\chi^2(3) = 5.566, p = 0.135\). We conclude that hypothesis 9 is refuted.

**H10: Values for the HR are different between game versions.**

The Shapiro-Wilk test was significant for the data generated after playing versions A \((D(29) = 0.915, p < 0.05)\) and D \((D(29) = 0.908, p < 0.05)\). Versions B \((D(29) = 0.930, p = 0.054)\) and C \((D(29) = 0.945, p = 0.138)\) were not significant. There was no significant difference between the HR values corresponding to the different versions of the game, \(\chi^2(3) = 2.255, p = 0.521\). We refute hypothesis 10.

**H11: Values for the GSR are different between game versions.**

The Shapiro-Wilk test was significant for the data generated after playing versions A \((D(29) = 0.915, p < 0.001)\), B \((D(29) = 0.930, p < 0.001)\), C \((D(29) = 0.945, p < 0.001)\) and D \((D(29) = 0.908, p < 0.001)\). There was no significant difference between the GSR values corresponding to the different versions of the game, \(\chi^2(3) = 4.972, p = 0.174\). Therefore, hypothesis 11 is refuted.

Findings about the alpha band are supported by Chanel et al. [3] given that the alpha band was not significant in the distinction between the version the player was playing. Similarly, our findings on the beta band are in accordance with Berta et al. [25] regarding the beta band allowing to distinguish the flow state from the boredom and frustration. Lastly, the theta band findings are not supported by Berta et al. [25] or by Chanel et al. [3], in spite of the significance value being relatively low.

Our results show that no single brainwave band provides enough information to discriminate between all four different game versions. Only the beta band proved to have significant differences, comparing to the alpha and theta bands. We can conclude that this is the best band of brainwaves to use when we are trying to distinguish the mental state of the player.

Analysis for the HR and GSR also did not provide any measure that could effectively differentiate
signal measures from the four versions. We discarded the usage of GSR to detect variations in the mental state because, as mentioned in Section 4.2.1, the appearance of missing values was too common to provide a reliable analysis.

Since we did not aim to create a game which could only adapt to the player based on one physical feature, we decided to further analyze the values of the alpha and theta bands and the HR. We decided to follow-up with Wilcoxon tests to see how significant were the differences in these three signals between the versions. We put together all the significance values in table 4.3 to compare which of these three physical measures could better complement the beta band.

Table 4.3: Asymptotic significance (2-tailed) values between alpha and theta bands and HR for each pair of versions.

<table>
<thead>
<tr>
<th></th>
<th>Boredom-Anxiety</th>
<th>Engagement-Anxiety</th>
<th>Frustration-Anxiety</th>
<th>Engagement-Boredom</th>
<th>Frustration-Boredom</th>
<th>Frustration-Engagement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha band</td>
<td>0.721</td>
<td>0.456</td>
<td>0.325</td>
<td>0.510</td>
<td>0.325</td>
<td>0.552</td>
</tr>
<tr>
<td>Theta band</td>
<td>0.230</td>
<td>0.275</td>
<td>0.347</td>
<td>0.381</td>
<td>0.417</td>
<td>0.922</td>
</tr>
<tr>
<td>HR</td>
<td>0.787</td>
<td>0.596</td>
<td>0.294</td>
<td>0.940</td>
<td>0.347</td>
<td>0.133</td>
</tr>
<tr>
<td>Best measure</td>
<td>Theta</td>
<td>Theta</td>
<td>HR</td>
<td>Theta</td>
<td>Alpha</td>
<td>HR</td>
</tr>
</tbody>
</table>

Since we could already distinguish between the pairs Engagement-Boredom and Frustration-Boredom, we analyzed the other four. Both theta band and HR could help distinguishing two other combinations each. The alpha band was better just to distinguishing the Frustration-Boredom pair (and beta band was much better doing so). In order to decide between the theta band and the HR, we considered that, since frustration happens at the limit of anxiety, we could assume anxiety and frustration as an hypothetical same mental state. Therefore, we were already able to distinguish this mental state from boredom (Frustration-Boredom pair) and flow from boredom (Engagement-Boredom pair) with the beta band. We were missing a way to distinguish anxiety or frustration from engagement. To distinguish that hypothetical mental state from engagement, we compared the significance of the combinations Engagement-Anxiety and Frustration-Engagement. The one with the lowest significance was the Frustration-Engagement (0.275 > 0.133), so we decided to use the HR as the second physiological measure. There was no need to use the alpha or theta bands as well, since they would not introduce any more relevant information.

4.2.2 Classification Algorithm

This section presents the modeling of the classifier. It starts by explaining how and which features we chose to work with the beta band and HR recordings. Using those features we test four different classifiers: Decision Tree (DT), Random Forest (RF), SVM and Multilayer Perceptron (MLP). Finally, we discuss results and choose which classifier to use in the final solution.

The features extraction and the modelling of the classification algorithm were programmed in Python.
2.7 and run with the Anaconda platform\textsuperscript{10}.

Before starting to discuss the features, we must take some considerations regarding how we treated data before feeding it to the classifier.

First, we decided to label recorded data from each version as the dominant mental state the player felt. In order to achieve this, we compare the scores on the anxiety, boredom, frustration items and Flow Degree value for each version. From these four values, the highest is the mental state that we associate with the physiological data recorded for that version. As an example, imagine that a participant plays version A. The values he notes down in the items and we calculate are the following:

- "I felt anxious" = 5;
- "I felt bored" = 1;
- "I felt frustrated" = 7; and
- Normalized Flow Degree = 4.

Therefore, we do not label physiological data recorded while the tester was playing that version as "Anxiety". The highest value was the item regarding the frustration, so we label that data as "Frustration". This method allows us to create a classifier that is adjusted to what the player felt in each game and not to what they was supposed to feel.

Second, as we mentioned before, frustration emerges at the limit of anxiety. Therefore, and taking into account that it would simplify the problem, we decided to merge the anxiety and frustration mental states. This way we have to address only three mental states (anxiety, boredom and flow) and the transition between these three can be easily set by adjusting the difficulty based on how the classifier reads the mental state.

Third, we used the difference between the values measured while the participant was playing the game and the mean of the control record. These allows us to create a classifier that addresses every user instead of a specific one, since it uses the differences between mental states rather than the real values.

Summing up, before we started to extract features, we had all the recorded data labeled as "Anxiety", "Boredom" or "Flow" based on the dominant mental state the participants felt while we recorded their physical signals.

4.2.2.1 Choosing Features

Before choosing features, we started by choosing the time interval in which the classifier would check the mental state of the user. Through informal contact, users claimed that twenty seconds is enough

\textsuperscript{10}www.continuum.io/anaconda-overview
time to change their mental state while they played the game. Therefore, we decided to adopt that time length to check the mental state of the user, i.e. at each twenty seconds, the classifiers evaluates the mental state of the user based on the last twenty seconds and outputs it.

The number of dimensions of our data is ten, five per each signal. Two features we decided to use are the mean and its variance. We chose them due to the fact that their combination can represent the shape of a distribution. The other three features are based on a Fast Fourier Transform (FFT). Each of them is composed by the sum of all real values that appear inside a certain frequency range. The intervals were chosen based on the shape of the real values of the FFTs and that artifacts would not be in the center of the graph. Figures 4.32 and 4.33 show an example on how the areas were divided for beta and HR signals, respectively. We decided to give an identifier to each feature, so that we can easily address them. See Table 4.4 for further details.

Figure 4.32: FFT feature areas for beta band. The three different colored areas represent the three different frequency ranges. The feature represented by the green color sums all the values that are in the interval $[-0.5;0.5]$. The orange one is the sum of all values that are in $[-1;-0.5] \cup [0.5;1]$. Finally, the feature associated with the red color is the sum of all values in $[-1.5;-1] \cup [1;1.5]$.

Before starting to model classifiers, we had to be sure that our features were adequate. First, we analyzed how they were distributed with histograms (see Figure 4.34). We can observe that data is variably distributed between all features with a low number of outliers.

We also created a Pearson correlation heatmap (see Figure 4.35). It shows that there is a medium correlation in the feature pairs (HR_AVG_20s, HR_FFT_0_05) and (BETA_AVG_20s, BETA_FFT_0_5). These correlations were expected since the average of the signals would be similar to the sum of the most frequent values from the distribution. We have to add that, since classifiers have difficulties in
Figure 4.33: FFT feature areas for HR. The three different colored areas represent the three different frequency ranges. The feature represented by the green color sums all the values that are in the interval \([-0.05;0.05]\). The orange one is the sum of all values that are in \([-0.1;-0.05] \cup [0.05;0.1]\). Finally, the feature associated with the red color is the sum of all values in \([-0.15;-0.1] \cup [0.1;0.15]\).

Table 4.4: Identifiers for each feature.

<table>
<thead>
<tr>
<th>Feature ID</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA_AVG_20s</td>
<td>Mean of the beta band</td>
</tr>
<tr>
<td>BETA_VAR_20s</td>
<td>Variance of the beta band</td>
</tr>
<tr>
<td>BETA_FFT_0_5</td>
<td>Sum of real beta band FFT values in the frequency interval ([-0.5;0.5])</td>
</tr>
<tr>
<td>BETA_FFT_1_0</td>
<td>Sum of real beta band FFT values in the frequency interval ([-1;-0.5] \cup [0.5;1])</td>
</tr>
<tr>
<td>BETA_FFT_1_5</td>
<td>Sum of real beta band FFT values in the frequency interval ([-1.5;-1] \cup [1;1.5])</td>
</tr>
<tr>
<td>HR_AVG_20s</td>
<td>Mean of the instantaneous HR</td>
</tr>
<tr>
<td>HR_VAR_20s</td>
<td>Variance of the instantaneous HR</td>
</tr>
<tr>
<td>HR_FFT_0_05</td>
<td>Sum of real HR FFT values in the frequency interval ([-0.05;0.05])</td>
</tr>
<tr>
<td>HR_FFT_0_10</td>
<td>Sum of real HR FFT values in the frequency interval ([-0.1;-0.05] \cup [0.05;0.1])</td>
</tr>
<tr>
<td>HR_FFT_0_15</td>
<td>Sum of real HR FFT values in the frequency interval ([-0.15;-0.1] \cup [0.1;0.15])</td>
</tr>
</tbody>
</table>
Figure 4.34: 20-bin histograms with the distribution of the chosen features.
distinguishing between states if the features are highly correlated, we can observe that the other correlations present minimal or small values between the features. We conclude that the chosen features will be efficient to distinguish the mental state of the user.

Figure 4.35: Pearson correlation heatmap of chosen features and the mental state label.

4.2.2.2 Choosing a Classifier

With our features defined, we tested several classifiers. We chose to test four classification algorithms: DT, RF, SVM and MLP. Firstly, we randomly changed the order of the lines from the data set so that we can have an even label distribution among the whole set. After that, we took 20% of the data to use as the test set for all the classification algorithms. Then, we normalized data by mean and variance. We used the scikit-learn toolkit\textsuperscript{11} to process data and model the classifiers. This section covers how we modeled each of the four classifiers using a 10-fold cross-validation grid search method with a set of hyper-parameters. This fine tuning technique lets us increase, in our case, the accuracy with

\textsuperscript{11}http://scikit-learn.org/stable/
hyper-parameters that change the behavior of the classifier. We chose hyper-parameters ranges with values inside the usually used ranges and not in the extremes. This way we obtain the classifiers with the highest accuracy and its parameters. In the end of this section, we compare how each classifier fared against the test set and choose which one we use to address the mental state of the user.

Decision Tree

In the end of the grid search, the DTs had a mean accuracy of 0.7141 with a standard deviation of 0.0201 in the validation set. The highest accuracy with the validation set was 0.7190. Parameters for this DT are present in Table 4.5.

**Table 4.5:** Parameters of the decision tree with highest accuracy.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>min_samples_leaf</td>
<td>2</td>
</tr>
<tr>
<td>min_samples_split</td>
<td>7</td>
</tr>
<tr>
<td>random_state</td>
<td>1</td>
</tr>
<tr>
<td>max_features</td>
<td>sqrt</td>
</tr>
<tr>
<td>max_depth</td>
<td>29</td>
</tr>
<tr>
<td>class_weight</td>
<td>balanced</td>
</tr>
</tbody>
</table>

With the test set, the best DT had an accuracy of 0.72291. Tables 4.6, 4.7 and 4.8 show the confusion matrix, the classification report and the importance the best DT gave to each feature when testing with the test set, respectively. We can observe that the DT distinguished with almost the same precision all states, in spite of boredom only being approximately 10% of the data from the test set. Also, the sum of the importance of features regarding the HR is approximately 60.40% of the total. This confirms that our choice to include the HR in the measures to distinguish the player was good.

**Table 4.6:** Confusion matrix of decision tree with test set.

<table>
<thead>
<tr>
<th></th>
<th>Anxiety</th>
<th>Flow</th>
<th>Boredom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>232</td>
<td>60</td>
<td>6</td>
</tr>
<tr>
<td>Flow</td>
<td>72</td>
<td>218</td>
<td>4</td>
</tr>
<tr>
<td>Boredom</td>
<td>9</td>
<td>12</td>
<td>33</td>
</tr>
</tbody>
</table>

**Table 4.7:** Classification report of decision tree with test set.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>0.74</td>
<td>0.78</td>
<td>0.76</td>
<td>298</td>
</tr>
<tr>
<td>Flow</td>
<td>0.75</td>
<td>0.74</td>
<td>0.75</td>
<td>294</td>
</tr>
<tr>
<td>Boredom</td>
<td>0.77</td>
<td>0.61</td>
<td>0.68</td>
<td>54</td>
</tr>
<tr>
<td>avg / total</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>646</td>
</tr>
</tbody>
</table>
Table 4.8: Importance of each feature of decision tree with test set.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA_AVG_20s</td>
<td>0.10535228</td>
</tr>
<tr>
<td>BETA_VAR_20s</td>
<td>0.0682024</td>
</tr>
<tr>
<td>BETA_FFT_0_5</td>
<td>0.08391256</td>
</tr>
<tr>
<td>BETA_FFT_1_0</td>
<td>0.04356395</td>
</tr>
<tr>
<td>BETA_FFT_1_5</td>
<td>0.0949618</td>
</tr>
<tr>
<td>HR_AVG_20s</td>
<td>0.13615048</td>
</tr>
<tr>
<td>HR_VAR_20s</td>
<td>0.12711029</td>
</tr>
<tr>
<td>HR_FFT_0_05</td>
<td>0.14738186</td>
</tr>
<tr>
<td>HR_FFT_0_10</td>
<td>0.05253448</td>
</tr>
<tr>
<td>HR_FFT_0_15</td>
<td>0.1408299</td>
</tr>
</tbody>
</table>

Random Forest

The RF had a mean accuracy of 0.8428 with a standard deviation of 0.0129 in the validation set. The highest accuracy with the validation set was 0.8527. The best parameters are shown in Table 4.9.

Table 4.9: Parameters of the random forest with highest accuracy.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>warm_start</td>
<td>False</td>
</tr>
<tr>
<td>min_samples_leaf</td>
<td>2</td>
</tr>
<tr>
<td>n_estimators</td>
<td>1000</td>
</tr>
<tr>
<td>max_features</td>
<td>sqrt</td>
</tr>
<tr>
<td>random_state</td>
<td>1</td>
</tr>
<tr>
<td>min_samples_split</td>
<td>5</td>
</tr>
<tr>
<td>max_depth</td>
<td>33</td>
</tr>
<tr>
<td>class_weight</td>
<td>None</td>
</tr>
</tbody>
</table>

With the test set, the best RF had an accuracy of 0.8514, which is close to what it had with the validation set. Tables 4.10, 4.11 and 4.12 show the confusion matrix, the classification report and the importance the best RF gave to each feature when testing with the test set, respectively. Although the anxiety and flow states are high, the boredom had a much lower precision of 0.56, which explains the recall close to 100%, which is a very good value. When it comes to the importance of the features, this time they were approximately divided with 50% for each beta band and HR.

Table 4.10: Confusion matrix of random forest with test set.

<table>
<thead>
<tr>
<th></th>
<th>Anxiety</th>
<th>Flow</th>
<th>Boredom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>279</td>
<td>43</td>
<td>14</td>
</tr>
<tr>
<td>Flow</td>
<td>33</td>
<td>247</td>
<td>5</td>
</tr>
<tr>
<td>Boredom</td>
<td>1</td>
<td>0</td>
<td>24</td>
</tr>
</tbody>
</table>
Table 4.11: Classification report of random forest with test set.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>0.89</td>
<td>0.83</td>
<td>0.86</td>
<td>336</td>
</tr>
<tr>
<td>Flow</td>
<td>0.85</td>
<td>0.87</td>
<td>0.86</td>
<td>285</td>
</tr>
<tr>
<td>Boredom</td>
<td>0.56</td>
<td>0.96</td>
<td>0.71</td>
<td>25</td>
</tr>
<tr>
<td>avg / total</td>
<td>0.86</td>
<td>0.85</td>
<td>0.85</td>
<td>646</td>
</tr>
</tbody>
</table>

Table 4.12: Importance of each feature of random forest with test set.

<table>
<thead>
<tr>
<th>Feature Importance</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA_AVG_20s</td>
<td>0.12103885</td>
</tr>
<tr>
<td>BETA_VAR_20s</td>
<td>0.09567193</td>
</tr>
<tr>
<td>BETA_FFT_0_5</td>
<td>0.10298836</td>
</tr>
<tr>
<td>BETA_FFT_1_0</td>
<td>0.09205427</td>
</tr>
<tr>
<td>BETA_FFT_1_5</td>
<td>0.08683688</td>
</tr>
<tr>
<td>HR_AVG_20s</td>
<td>0.10275214</td>
</tr>
<tr>
<td>HR_VAR_20s</td>
<td>0.08590153</td>
</tr>
<tr>
<td>HR_FFT_0_05</td>
<td>0.10874959</td>
</tr>
<tr>
<td>HR_FFT_0_10</td>
<td>0.0891089</td>
</tr>
<tr>
<td>HR_FFT_0_15</td>
<td>0.11489754</td>
</tr>
</tbody>
</table>

SVM

The best SVM classifier had a mean accuracy of 0.7058 with a standard deviation of 0.0949 in the validation set. The highest accuracy with the validation set was 0.8333. The best parameters are shown in Table 4.13.

Table 4.13: Parameters of the support vector machine with highest accuracy.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>kernel</td>
<td>rbf</td>
</tr>
<tr>
<td>C</td>
<td>1000</td>
</tr>
<tr>
<td>probability</td>
<td>True</td>
</tr>
<tr>
<td>max_iter</td>
<td>10000</td>
</tr>
<tr>
<td>random_state</td>
<td>1</td>
</tr>
<tr>
<td>gamma</td>
<td>0.1</td>
</tr>
<tr>
<td>class_weight</td>
<td>balanced</td>
</tr>
</tbody>
</table>

With the test set, the best SVM had an accuracy of 0.8297. Tables 4.14 and 4.15 show the confusion matrix and the classification report with the test set, respectively. The accuracy value with the test set is close to the one with the validation set, so we can argue that this classifier can not be further optimized with these features. The precision values regarding anxiety and boredom are relatively low, comparing to flow. Nonetheless, the average values from the report are inline with what we expected for this classifier.
Table 4.14: Confusion matrix of support vector machine with test set.

<table>
<thead>
<tr>
<th></th>
<th>Anxiety</th>
<th>Flow</th>
<th>Boredom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>256</td>
<td>39</td>
<td>7</td>
</tr>
<tr>
<td>Flow</td>
<td>54</td>
<td>247</td>
<td>3</td>
</tr>
<tr>
<td>Boredom</td>
<td>3</td>
<td>4</td>
<td>33</td>
</tr>
</tbody>
</table>

Table 4.15: Classification report of support vector machine with test set.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>0.82</td>
<td>0.85</td>
<td>0.83</td>
<td>302</td>
</tr>
<tr>
<td>Flow</td>
<td>0.85</td>
<td>0.81</td>
<td>0.83</td>
<td>304</td>
</tr>
<tr>
<td>Boredom</td>
<td>0.77</td>
<td>0.82</td>
<td>0.80</td>
<td>40</td>
</tr>
<tr>
<td>avg / total</td>
<td>0.83</td>
<td>0.83</td>
<td>0.83</td>
<td>646</td>
</tr>
</tbody>
</table>

MLP

We decided not to create a three hidden layers MLP since it usually leads to overfitting. With this in mind, the maximum number of layers we tried were two. The best MLP had a mean accuracy of 0.8631 with a standard deviation of 0.0183 in the validation set. The highest accuracy with the validation set was 0.8895. The best parameters are shown in Table 4.16.

Table 4.16: Parameters of the multilayer perceptron with highest accuracy.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>beta_1</td>
<td>0.9</td>
</tr>
<tr>
<td>beta_2</td>
<td>0.7</td>
</tr>
<tr>
<td>hidden_layer_sizes</td>
<td>(1000, 1000)</td>
</tr>
<tr>
<td>epsilon</td>
<td>1e-07</td>
</tr>
<tr>
<td>learning_rate</td>
<td>invscaling</td>
</tr>
<tr>
<td>max_iter</td>
<td>1000</td>
</tr>
<tr>
<td>batch_size</td>
<td>10</td>
</tr>
<tr>
<td>random_state</td>
<td>1</td>
</tr>
<tr>
<td>tol</td>
<td>1e-06</td>
</tr>
<tr>
<td>alpha</td>
<td>1e-05</td>
</tr>
</tbody>
</table>

With the test set, the best MLP had an accuracy of 0.8669. Tables 4.17 and 4.18 show the confusion matrix and the classification report with the test set, respectively.

Table 4.17: Confusion matrix of multilayer perceptron with test set.

<table>
<thead>
<tr>
<th></th>
<th>Anxiety</th>
<th>Flow</th>
<th>Boredom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>271</td>
<td>34</td>
<td>5</td>
</tr>
<tr>
<td>Flow</td>
<td>35</td>
<td>254</td>
<td>3</td>
</tr>
<tr>
<td>Boredom</td>
<td>7</td>
<td>2</td>
<td>35</td>
</tr>
</tbody>
</table>
### Table 4.18: Classification report of multilayer perceptron with test set.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anxiety</td>
<td>0.87</td>
<td>0.87</td>
<td>0.87</td>
<td>310</td>
</tr>
<tr>
<td>Flow</td>
<td>0.88</td>
<td>0.87</td>
<td>0.87</td>
<td>292</td>
</tr>
<tr>
<td>Boredom</td>
<td>0.81</td>
<td>0.80</td>
<td>0.80</td>
<td>44</td>
</tr>
<tr>
<td>avg / total</td>
<td>0.87</td>
<td>0.87</td>
<td>0.87</td>
<td>646</td>
</tr>
</tbody>
</table>

### Discussion

After modeling the four different classifiers, we had to choose which one we would use in our game. Table 4.19 shows all the relevant values in which we base our choice.

### Table 4.19: Comparison of the classifiers with average test set results.

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
<th>Sup. Anxiety</th>
<th>Sup. Flow</th>
<th>Sup. Boredom</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>0.72</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>298</td>
<td>294</td>
<td>54</td>
</tr>
<tr>
<td>RF</td>
<td>0.85</td>
<td>0.86</td>
<td>0.85</td>
<td>0.85</td>
<td>336</td>
<td>285</td>
<td>25</td>
</tr>
<tr>
<td>SVM</td>
<td>0.83</td>
<td>0.83</td>
<td>0.83</td>
<td>0.83</td>
<td>302</td>
<td>304</td>
<td>40</td>
</tr>
<tr>
<td>MLP</td>
<td>0.87</td>
<td>0.87</td>
<td>0.87</td>
<td>0.87</td>
<td>310</td>
<td>292</td>
<td>44</td>
</tr>
</tbody>
</table>

The MLP showed higher accuracy, precision, recall and f1-score compared to the other three classifiers, although the differences against the RF and SVM are not significant. One reason why it happened may be related to the fact that we only use ten features and the classifiers reached a maximum value for all metrics with our dataset. Regarding support, RF has the higher value for anxiety, SVM for flow and DT for boredom. Although MLP does not have a higher support for the mental states, no other classifier dominates these three last measures as the MLP dominates the first four. This leads us to conclude that the best classifier for our work is the MLP.

#### 4.2.2.3 Discussion

This section presented one of the core pieces of the final framework to detect the mental state of the participant. First, we processed data in order to be able to take our features from the beta band and the instantaneous HR. Next, we chose the features for each physiological signal that we decided were the best for our problem: average, variance and three sums of specific ranges from a FFT of each twenty seconds the participant was playing. Using our features, we modeled four different classifiers: DT, RF, SVM and MLP. The one with the best general results was the MLP, so this will be the chosen classifier in our framework.
4.2.3 Architecture

Figure 4.36 depicts the final architecture of the classification framework. The illustrated process occurs for time intervals of twenty seconds. At each twenty seconds, a bash script runs the BioSPPy toolkit and processes the text file from OpenSignals (r)evolution. BioSPPy outputs two text files: one has the instantaneous HR and the other the values for the beta band. Next, the bash script calls a Python scripts which extracts the features from those files, loads the classifier and writes on a file the classified mental state.

![Figure 4.36: Framework that classifies the mental state of the user. From left to right, it is composed by User, Bitalino, BioSPPy, Feature Extraction Algorithm and Classifier. The flow of information in this framework is straightforward. First, the User produces analog signals which are measured with the Bitalino. These signals are then digitized by OpenSignals (r)evolution and fed to the BioSPPy which filters them, performing R-peak detection for the computation of the instantaneous HR and EEG band division for the beta waves. After that, the Feature Extraction Algorithm takes both the beta band and the HR and extracts the relevant features, which are fed to the Classifier.](image)

4.2.4 Discussion

This section presented how the classification framework was created and how its final architecture is composed. In short, we started by choosing the best measures to differentiate the mental states from each others. After it, we processed the data so that we could extract relevant features and used them to model the classification algorithm. Finally, we have a functional framework that can detect a representation of the mental state of the player with an accuracy of 87%. We have now analyzed two of the three main contributions of our work. The next section will address the third component: the controller framework.

4.3 Controller Framework

The third and final main contribution is the controller framework. It is responsible for changing the parameters of the game based on the mental state of the player. In order to create a system that has to keep information regarding the current state, a list of possible states to transit to and a matrix of transitions, we opted for a state machine. Therefore, we implemented a state machine in our adaptable game which allows the game engine to know the current state and, when it receives the up-to-date mental state of the user, it transits to the next state depending on it. This section presents how the
state machine was developed and how the game conducts the adaptation of the game parameters and environment settings.

4.3.1 State Machine

We opted to create a game that has three different levels. The mental state of the player is the trigger that allows the gamer to travel between the different levels while they are playing. The transitions the state machine receives as input are the output of the classification algorithm. After the state machine updates the current state, the game adapts itself to it. We did not find that more states would give a better experience, since three were enough to represent changes in the gameplay.

The game has an object called Game Manager and its function is to regulate the game’s level. The next transition is written on a text file that the Game Manager reads each twenty seconds. That object has a script with the current state machine (see Figure 4.37) and is responsible for changing the gameplay depending on the level. It is similar to the one developed by Rani et al. [2].

We defined that the level 1 would be the state with the least engagement elements and the easiest in terms of difficulty. On the contrary, level 3 would have all the engagement elements and it would be the hardest. This way we can always keep the player in a flow state. As an example, if the difficulty is too high for their skills, they will become anxious; the state machine must decrease the game difficulty which is equivalent to go down one level. With a lower difficulty, players will reach a flow state. While their skills match the difficulty of that level, they will be in flow and developing their skills until they are too high for the challenge of that level. In that case, the player will be bored and the state machine must go one level up. With an increase of difficulty, they will reach flow again and continue on developing their skills. This process is the core of the game that lets players have the most enjoyable experience while they are playing the game because it tends to always leave them in a flow state.

![Figure 4.37: Three states machine diagram for mental state-based gameplay adaptation. The player’s mental state is classified as one of the following states: Anxiety, Boredom and Flow. After getting that transition, the state machine processes it and updates the state. When the game starts, the player starts at level 1. If their mental state is classified as Boredom, the player progresses to the second level of the state machine; otherwise, the player remains on the first level. If the player is on level 2, they go back to level 1 with an Anxiety classification, stay in level 2 with a Flow classification and jump to level 3 with a Boredom one. Finally, the player remains in level 3 as long as their mental state is Boredom or Flow; if it is Anxiety they go back to level 2.](image)
As we mentioned in Section 1.1, we use two prototypes in the testing phase. Besides the mental state-based state game, we require one that adapts to the player’s performance. Thus, we created a second version of the game with a state machine that adapts to the performance (see Figure 4.38).

![Three states machine diagram for performance-based gameplay adaptation.](image)

**Figure 4.38:** Three states machine diagram for performance-based gameplay adaptation. The player’s performance is classified as one of the following cases: Poor, Medium and Excellent. At each specific time interval, the game checks how was the performance in the last time interval and feeds it to the state machine. After getting that transition, the state machine processes it and updates the state. When the game starts, the player starts at level 1. If their performance is classified as Excellent, the player progresses to the second level of the state machine; otherwise, the player remains on the first level. If the player is on level 2, they go back to level 1 with Poor, stay in level 2 with a Medium and jump to level 3 with an Excellent. Finally, the player remains in level 3 as long as their performance is Medium or Excellent; if it is Poor they go back to level 2.

With the number of levels defined, we needed to create the guidelines for the adaptation. The design criteria to keep the user in a flow state are concentrated on two different modules: environmental settings to adapt engagement and enemies stats alteration to adapt the difficulty.

### 4.3.2 Environmental Settings

As mentioned before, we needed to create elements in the game that would lead the player to a higher engagement and, consequently, to flow. The design guidelines for the environmental settings are the following:

- In levels 2 and 3, when enemies die they explode giving AOE damage to nearby enemies;
- In level 3, barrels spawn randomly on the environment and give AOE damage to nearby enemies when they explode;
- Color of explosions depends on the flow of the player; and
- Fitting sensory effects (explosions, sounds, textures).

With these guidelines, we change the way the player interacts with the enemies thereby creating engagement. The objective is not to prolong the task of killing an endless wave of zombies, since that can lead to boredom. This way, the player can create new strategies based on the level. The differences in the color of explosions is a passive way of showing the player that the level changed.
These guidelines led to a further development of our game. We decided to reuse the explosion particle system developed for the explosion of a rocket. As mentioned before, enemies explode when they die and give AOE damage to surrounding enemies, as can be seen in Figure 4.39.

Pictured in Figure 4.40, the barrels are a model with high contrast colors to allow players to easily distinguish them from the rest of the game objects. We decided to add them to the game since they allow the players to create more strategies while dealing with the enemies, thus improving engagement. The barrels have a spawn similar to the pick-ups. There is a set of positions in which a barrel can spawn and there can only be one barrel in each position at a time. When the player shoots a barrel or when a barrel is in the range of an explosion, it instantiates an explosion equal to the one the rockets produce.

4.3.3 Difficulty Settings

Regarding the difficulty adaptation, as higher the level, the difficulty is also harder. We decided that enemies gain speed, health and shorter spawn rate as the player goes up in levels, gradually increasing combat difficulty gradually. The guidelines presented in the previous section allow the player to fight harder enemies, since killing one or shooting at a barrel allows the player to deal damage to any number of surrounding enemies. This balance is responsible for creating a higher engagement although the enemies are harder to kill.

The stats differences (see Table 4.20) between levels were taken from the results in the first testing phase. We decided that level 1 would have the stats of the engagement version and level 3 of the anxiety version. Level 2 has values for the stats between levels 1 and 3 to create a sense of progression. Since the engagement version had a higher level of Flow Degree, we expect that it can counter the lack of engagement elements in level 1. The only differences are in the pick-ups spawn time and probabilities. We decided to give a longer time length in level 1 because we also increased the ammo probability. This way, the player has a higher probability of acquiring ammo, but it doesn’t spawn as fast as before. We believe that it leads players to develop a more attacking strategy to leave them more engaged in the game. We always keep the same rocket ammo probability because informal contact in the first testing phase claimed that it was one of the most engaging pieces of the game. Therefore, players aim to
Table 4.20: Stats differences between levels.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zombunny Spawn Time</td>
<td>3</td>
<td>2.5</td>
<td>2</td>
</tr>
<tr>
<td>Zombunny Speed</td>
<td>5.5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Zombunny Health</td>
<td>60</td>
<td>50</td>
<td>40</td>
</tr>
<tr>
<td>Zombear Spawn Time</td>
<td>4</td>
<td>3.5</td>
<td>3</td>
</tr>
<tr>
<td>Zombear Speed</td>
<td>4.5</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Zombear Health</td>
<td>100</td>
<td>90</td>
<td>80</td>
</tr>
<tr>
<td>Hellephant Spawn Time</td>
<td>10</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>Hellephant Speed</td>
<td>2.5</td>
<td>2.5</td>
<td>3</td>
</tr>
<tr>
<td>Hellephant Health</td>
<td>340</td>
<td>290</td>
<td>240</td>
</tr>
<tr>
<td>Pick-Ups Spawn Time</td>
<td>3</td>
<td>2.5</td>
<td>2</td>
</tr>
<tr>
<td>Gun Ammo Probability</td>
<td>40%</td>
<td>35%</td>
<td>30%</td>
</tr>
<tr>
<td>Rocket Ammo Probability</td>
<td>40%</td>
<td>40%</td>
<td>40%</td>
</tr>
<tr>
<td>Health Probability</td>
<td>20%</td>
<td>25%</td>
<td>30%</td>
</tr>
</tbody>
</table>

...overuse it and the 40% probability lets them catch a good amount of rockets.

4.3.4 Discussion

This section illustrated the behavior of the controller framework in the context of our adaptable game. Besides explaining what kind of changes in parameters and environmental settings are applied in the game at each mental state update, it explained how the two prototypes deal with their metrics (mental state and performance in-game). With all the three main components fully explained, the next chapter covers the testing phase when we check which of the prototypes provides a better gaming experience.
5

Validation

Contents

5.1 Architecture ......................................................... 73
5.2 Method ................................................................. 74
5.3 Results .............................................................. 76
5.4 Discussion ............................................................ 81

71
The fifth chapter presents the validation of our work. It starts by showing the architecture of the solution and how its components interact among themselves. After mapping each element of the solution, the chapter summarizes the second testing phase of this work and presents the results. It ends with a discussion on the results we obtained.

5.1 Architecture

Depicted on Figure 5.1 is the architecture of the solution. There are five main components: the Player, the Classification Framework, the Controller Framework, the Game Engine and the Devices.

![Figure 5.1: Architecture of the flow-based adaptable gameplay prototype. The Player is represented by the icon inside the orange box (A). Inside the green box (B) are the components of the Classification Framework. The red box (C) has the Controller Framework and the blue one (D) has the Game Engine. Finally, the purple box (E) has all the devices the player interacts with in order to play the Game Engine (from top to bottom, Keyboard, Screen and Headphones). Arrows and their labels represent the meaningful information transmitted between the components.](image)

The Player component is the player themself and, while they are playing the game, they give an active input through the Keyboard, i.e. the player consciously coordinates their hands and fingers to perform some in-game actions so that they achieve some objective, and a passive input via the Bitalino, i.e. the device is reading the players’ physiological signals without they consciously perceiving the measurement. There are two ways by which the player perceives the game and they will be responsible for changing the user’s engagement state: the Headphones will take care of the audio input and the Screen of the visual one.

The Devices are the artifacts that interact with the player in an active or a passive way. They are the Keyboard (including a mouse) that sends the player’s conscious inputs to the Game Engine, the Screen and the Headphones. They represent the interface between the Player and the Game Engine.

As mentioned in Figure 4.36, the Classification Framework component has its four components: Bitalino, BioSPPy, Feature Extraction Algorithm and Classifier. The Classification Framework takes the
analog BVP and EEG signals as input and processes those signals in order to find the player's mental state. It outputs the mental state of the player corresponding to the last time frame.

The Controller Framework has the state machine responsible to control the game levels depending on the previous state and the up-to-date mental state of the player.

Finally, the Game Engine is the base game with all the adaptable parameters. These parameters are adjusted depending on the current level of the state machine from the Controller Framework.

In short, in order to play the game as it is, players need two main artifacts: a Bitalino and a computer. The Bitalino (with its components) measures the physiological signals which has three EEG channels (two prefrontal and one on the neck) and a pulse sensor. These sensors allow to measure both BVP and EEG. Players also need to have a computer with a screen, audio hardware, a keyboard and a mouse so that they can interact with the game. Headphones are optional, but using them allows a greater immersion. The computer is also responsible for running the classification framework.

5.2 Method

In this section, we describe the population sample used in our study, material we used in the tests and the procedure used in our second testing phase.

5.2.1 Participants

We recruited anyone interested in participating with at least 18 years old through standard procedures, including direct contact and through word of mouth. Again, participants were asked to sign a consent form and the experience presented no potential risks and no anticipated benefits to them.

We conducted a total of 21 tests. All tests occurred between 08:00h and 20:00h. There were no technical problems this time, so we kept a total of 21 completed tests for the analysis. The participants (16 males, 5 females) were ranged in age from 19 to 27 ($M = 22.43, SD = 1.91$). From all the participants, eleven had already been present in the first testing phase.

Only one participant reported no video game-playing time. The other participants play at least once a day (28.57%), at least once week (42.86%) or at least once a month (23.81%). Fifteen participants frequently play FPS. Again, a mixed ANOVA showed that playing in our first testing phase, gaming periodicity and gender had no significant effects on results.

5.2.2 User Evaluation

This testing phase had a new consent form and we decided to keep the PANAS [69] and adapted GEQ [48]. The participant filled three PANAS: one before playing any of the gaming versions, one in between them and one after playing them. This way we can address how each version changed the
participants’ positive and negative affects. The adapted GEQ was used to address the Flow Degree and see which version led to a higher self-perceived flow.

### 5.2.3 Apparatus

We used the same material mentioned in Section 4.1.5.3 (a Bitalino, a computer and their components) plus another computer. One technical limitation in our work stands on the OpenSignals (r)evolution software. It does not save on a file the physiological data while it is recording it. The assistant has to restart the recording each twenty seconds so that classification framework has a file to process. This means that we need two computers: the participant is playing in one and the assistant is running the classification framework in the other. Players interacted with the game through mousepad and headphones. We used the same set of images from Figures 4.23 and 4.24.

As mentioned before, the game needs to read the next state machine transition from a text file. This text file is produced by the classification framework. Since the game and the framework are in separate computers, we used a cloud service to keep a synchronized file between the two computers. That way, the framework writes the next transition in a text file inside the cloud service and, since it is only 1 byte, the syncing happens in milliseconds. The game that is running in the other computer reads this file and proceeds as we designed it. Other materials used for the experiences were a headband, neurodiagnostic electrode paste, alcohol and cloth.

### 5.2.4 Procedure

The tests were conducted in the same laboratory and in the same temperature interval. The assistant started by explaining to the users the purpose of the study, what they would be doing and that they should not move their head while they were playing in order to prevent detaching the electrodes. After it, we asked the users to fill a consent form for this testing phase and a new form regarding their demographic data, gaming experience and emotional state with PANAS. Again, the order of the adapted GEQ items was randomized between the two versions.

After filling it, we placed the physiological sensors on the user. The placement was the same as explained in Section 4.1.5.3, except this time we only placed the photoplethysmography sensor for the BVP and the three electrodes for the EEG. We also recorded the baseline values for five minutes while players looked at pictures from Figure 4.24. After that, users played a sandbox version of the game to try the sensitivity and in-game interactions. The assistant lets users play as much time as they wanted so that they could develop the minimum skills to play the game in the different versions. Even users that had participated in the first testing phase had to play to ensure that participants knew and remembered all in-game interactions.

Since results from the first testing phase showed that players would have more negative affect after
playing so much, we decided to cut down the length of the gaming sessions from forty to twenty minutes. Each tester played each version for ten minutes and the playing order was again randomized between players, meaning that players would either play first the mental state-based adaptable game and after it the performance-based adaptable game or vice-versa.

The procedure used for each version the participant played was the following:

1. The tester filled a PANAS scale;
2. The assistant verified if every sensor was correctly placed;
3. The assistant asked the tester if they was comfortable and ready to play;
4. The assistant started the game for the tester;
5. If the player died, the game automatically restarted;
6. The player played for ten minutes and after it the game automatically restarted;
7. The assistant asked the tester to fill the form addressing the version the latter played;
8. The player rested for three minutes looking at Figure 4.23 in order to return to a neutral mental state; and
9. After three minutes, the assistant repeated this procedure to the next version, if there was any other version to play.

We kept the game automatically restarting for the same reasons as in the first phase. After the tester played the two versions, the assistant removed the sensors from the tester and cleaned them with alcohol to remove the electrode gel. When the cleaning was done, the assistant asked them to fill another form with a PANAS scale. Free comments were also invited.

Testers received a compensation based in candies. Again, there was also a contest to see which player achieved the highest score across all gaming sessions. The winner received a gift card worth 20EUR.

5.3 Results

This section presents the results from our study. Sections 5.3.1 and 5.3.2 address all the useful information that we could extract from the questionnaires, namely how the players positive and negative affect varied and the self-perceived flow. Section 5.3.3 presents results regarding the performance of the participants using both versions of the adaptable game.

We always started by performing Shapiro-Wilk normality tests to decide which follow-up test to use. Follow-up tests were used to address if there were significant differences between the two versions in the
positive and negative affects of the player. This allows us to check if players liked the mental state-based adaptable game the most. Also, follow-up tests let us see significant differences in the self-perceived flow and scores between both versions. Thus, we can compare those results against the objectives of this work. The tests presented in the following sections are the follow-up tests to assess those statistical significant differences in data.

5.3.1 PANAS

From the PANAS we can conclude how each version affected the player, more precisely, which game they liked the best. Since the mental state-based adaptable game was created to provide a better gaming experience, we expect it to create a greater positive affect on the player compared to the performance-based adaptable game. This leads us to create the following hypothesis:

H12: Users have a higher positive affect playing the mental state-based version compared to the performance-based.

We can also expect that the version that will have a higher negative affect is the performance-based one, since it was not meant to provide an optimal experience. Therefore, we create another hypothesis:

H13: Users have a lower negative affect playing the mental state-based version compared to the performance-based.

H12: Users have a higher positive affect playing the mental state-based version compared to the performance-based

Shapiro-Wilk test was not significant for data regarding the performance-based (D(21) = 0.966, p > 0.05) or the mental-state based games (D(21) = 0.960, p > 0.05). We compared means from the positive affect between the mental state-based and the performance-based versions using a paired-samples t-test. There was no significant difference in the positive affect for the performance-based (M = 30.05, SD = 9.70) and mental state-based (M = 29.14, SD = 9.60) versions; t(20) = 1.133, p = 0.271. These results suggest that the version the participant played does not have an effect on their positive affect. Thus, the test for this hypothesis is inconclusive. Also, we observe in Figure 5.2 that minimum, maximum and IQR values are similar to both conditions, thus the variation in both conditions is similar. We need to address a larger number of participants to validate this hypothesis.

H13: Users have a lower negative affect playing the mental state-based version compared to the performance-based

Shapiro-Wilk test was significant for data regarding the performance-based (D(21) = 0.848, p < 0.05) and the mental-state based games (D(21) = 0.762, p < 0.001). The Wilcoxon test’s z-score, based on
the positive ranks, is $-2.029$ and this value is significant at $p = 0.042$. Therefore, it appeared that the degree of negative affect of the testers playing the performance-based ($Md_n = 13.45$) was significantly different compared to after the mental state-based ($Md_n = 12.52$), $T = 17.00, p < 0.05, r = -0.313$. We accept this hypothesis. Moreover, Figure 5.3 depicts the distribution and we observe that the mental state-based has an IQR with considerably lower values compared to the performance-based.

5.3.2 Flow

One of the objectives of this work is to prove that the self-perceived flow is higher when the user is playing a game that adapts to their mental state comparing to playing a game which gameplay adapts to their performance. In order to make this comparison, we will use the Flow Degree Scale we previously designed. We obtained data for the Flow Degree from the Adapted GEQ. We can then create another hypothesis:

**H14:** Users have a higher self-perceived flow playing the mental state-based version compared to the performance-based.
H14: Users have a higher self-perceived flow playing the mental state-based version compared to the performance-based

Shapiro-Wilk test was not significant for data regarding the performance-based ($D(21) = 0.977, p > 0.05$) and the mental-state based games ($D(21) = 0.967, p > 0.05$). We conducted a paired-samples t-test to compare the self-perceived flow using the normalized Flow Degree values between the mental state-based and the performance-based versions. There was a significant difference in the self-perceived flow for the performance-based ($M = 6.46, SD = 1.29$) and mental state-based ($M = 6.05, SD = 1.34$) conditions; ($t(20) = 2.215, p = 0.039$). These results show that the version the participant played has an effect on their self-perceived flow, more specifically that the performance-based prototype leads to a greater self-perceived flow compared to the mental state-based one. Therefore, we refute our hypothesis. Although both conditions have a similar IQR (therefore, a similar variation), we can observe in Figure 5.4 that the median of the performance-based version has a slightly higher value compared to the mental state-based version.

5.3.3 In-Game Scores

Lastly, other objective was to prove that a player has a higher performance when the user is playing a game that adapts to their mental state compared to playing a game which gameplay adapts to their
performance. We decided to compare this performance by the sum of all scores players would have by the end of each time they played. When an enemy was killed it provided a certain amount of points to the player. If it was a Zumbunny or a Zombear, the player would win 10 points; and 50 points if it was an Hellephant. We summed all the scores each player obtain while playing each version and created the last hypothesis:

**H15**: Users have higher scores playing the mental state-based version compared to the performance-based.

**H15**: Users have higher scores playing the mental state-based version compared to the performance-based

Shapiro-Wilk test was not significant for data regarding the performance-based ($D(21) = 0.962, p > 0.05$) and the mental-state based games ($D(21) = 0.939, p > 0.05$). A paired-samples t-test was conducted to compare the scores in the mental state-based and the performance-based versions. There was a significant difference in the scores for the performance-based ($M = 4820, SD = 1828.52$) and mental state-based ($M = 4326.67, SD = 1376.35$) conditions; $t(20) = 2.635, p = 0.016$. These results show that the version the participant played has an effect on their score, more specifically that users reach higher scores playing the performance-based prototype compared to the mental state-based one. Therefore, we refute our hypothesis. Additionally, Figure 5.5 depict the distributions and we see that
the performance-based version has a higher median compared to the mental state-based version. Yet, the mental state-based condition has a much smaller IQR which suggests that the variation was smaller compared to the one of the performance-based condition.

Figure 5.5: Boxplots of scores for performance and mental state-based versions.

5.4 Discussion

There are some important factors that may explain our results. One reason why this happened may be based on the limit values we used in the performance-based state machine. Each level has its own definition for Poor, Medium and Excellent performance in it. In level 1, players had an Excellent performance if they killed at least 75% of the enemies that spawned in the last twenty seconds. We choose 75% to make it hard to jump from levels 1 to 2, since in level 1 players only had access to the regular weapons and no other interactions to kill enemies. We used a similar approach in level 2, where participants only have an Excellent performance if they kill 90% of the enemies. The value is higher because it is easier to kill them, since enemies now explode when they die, so naturally they have to kill a greater number to go from levels 2 to 3 than from 1 to 2. Also, we have to consider that in level 2 enemies spawn faster than in level 1. With this technique, we aim to prevent players from climbing the levels too fast. If they went up levels too fast, they might end with mismatched skills for the challenge and die in a short time. We considered that it would ruin the gaming experience and players wouldn’t
be motivated to play well if they would end up dying quickly each twenty seconds. On the contrary, we decided to make it easier for the player to go down on levels. The principle is that, after players experiment the advantages of levels 2 and 3, they will focus on improving their performance to reach the next level. This way, if players are in level 2 and they kill less than 50% of the enemies or if they are in level 3 and kill less than 75% of the enemies, they go to level 1 and 2, respectively. Our first impression is that the values we used to define the Poor, Medium and Excellent were so well adjusted to lead the player to a higher level with more engagement elements that they ended up preferring the performance-based state machine instead of the one that tried to provide the best gaming experience by keeping a balance in challenge and skills.

The fact that our hypotheses were not accepted may be connected to some methodological limitations on our research that must be considered. When we collected data to model the classifier, the number of participants was small. A higher number of participants would allow to generalize for the whole population and take conclusions with a stronger impact. In future studies, a bigger sample size must be acquired. Also, the number of female participants was too small compared to the number of male participants, so future samples should consider increase female presence as well. These problems also showed up in the second testing phase, but with a smaller sample of participants and another disparate ratio of male and female gamers.

Moreover, our participants from the first testing phase did not cover the whole range of player expertise, as most players did not play the game at least once a week. Only in the first testing phase we found that the game periodicity influenced the anxiety in version C, yet, as it was only one case in sixteen possible cases, different gaming periodicity regarding FPS was assumed to be minimal. Concerning the second testing phase, the fact that the user played our game before or the game periodicity did not provide significant changes in the results.

Another limitation was the OpenSignals (r)evolution software. Since it did not allow a real-time recording, the assistant had to keep on restarting the recording after each twenty seconds. This was not practical and led to considerable small delays on the update of the file being processed by the classification framework. Nonetheless, these delays were rare or no longer than forty seconds, thus we consider them irrelevant.

Finally, we had to repeat the tests from the second testing phase because one of the sensors was not working properly and we only noted it after doing some signal processing.

Our findings provide additional evidence for inducing and investigating different mental states achieved while playing computer games. In particular, game developers may use the game design options to induce anxiety, boredom, flow and frustration or to maintain an optimal challenge based on the players mental state or performance. These mental states can be useful to provide a better gaming experience with task difficulty adjustment.
Conclusions and Future Work
This dissertation addresses the problem of these players becoming uninterested for not having their skills paired with the challenge of the game. In particular, we investigated if the mental state flow may be relevant for gameplay adaptability and may offer a better gaming experience, since that mental state is associated with deep engagement. In order to do so, we created an hypothesis supporting that a game that adapts to the gamer’s mental state following the flow theory provides a better gaming experience compared to a game that adapts to their performance. Although there were empirical evidences that users would have a higher flow state and scores playing a game that adapted itself to their mental state, after our testing phase, we found out that they have a higher flow state and scores playing a game that adapts to their performance. A deeper statistical analysis confirmed that players have a higher flow state and scores with a performance-based adaptable game, so in fact players do not have a higher flow state and scores with a mental state-based adaptable game. Hypotheses H14 and H15 addressed these topics and we refuted both, showing that participants had higher self-perceived flow and in-game performance in the performance-based adaptable game compared to the mental state-based adaptable game.

Our developed work resulted in a set of three main contributions: the game with adaptable parameters and environment, a classification framework that effectively detects the mental state of the player through biofeedback measures and a controller framework that effectively adapts a game based on the mental state of the player. We successfully created a classification framework that detects the player’s mental state in real time through biofeedback measures. Laine et al. [62] had an accuracy of 87% and Wilson and Russel [63] had mean classification accuracies of 85%, 82% and 86%. Our values are similar to their work, since the MLP had an accuracy as high as 89% with the validation set and 87% with the test set. We also contribute with an adaptable game design and the modeling of the controller framework. Both the game and the controller framework proved to effectively adapt the game engine depending on the feedback from the classification framework. We consider these to be relevant contributions to the field of study, which will lead to the writing and submission of another scientific paper.

In the end, we were not able to accomplish our main goal which was to prove that a mental state-based adaptable game provided a better gaming experience to the player. In spite of it, we need further research to study our hypothesis in full, since we had a small number of participants and without our desired FPS gaming periodicity.

Future work involves creating a larger data set with a higher number of participants of both genders. This way we can generalize for the whole population and model a classifier with higher accuracy. We can also validate the plots created in Section 4.1.3, so that we can extend them to all the FPS genre. The number of participants that play FPS must be increased, since they are used to the game type and can provide a better feedback than players who only play mobile games, for example. Another approach is to complement our dataset with more representations of mental states of the user in order to provide
a wider set of adaptability components and address more dimensions other than skills of the player and challenge of the game. Further development of the MLP classifier involves using different features from the ones we chose and choosing different physiological signals. Regarding software we used in our study, there is future work using the OpenSignals (r)evolution application programming interface to create a real-time recording of the physiological signals, which may prove to be a more effective way to access the participants’ physiological signals without the disruption of restarting the recording at each time interval.
Bibliography


