



## **Jogo Sérió para a Prevenção do Stress em Espaços Profissionais**

**CASSANDRA SOARES RANGINHA**

Outubro de 2022

# **Jogo Sérió para a Prevenção do Stress em Espaços Profissionais**

**Cassandra Soares Ranginha**

**Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática, Área de Especialização em  
Sistemas Gráficos e Multimédia**

**Orientador: Prof.<sup>a</sup> Dr.<sup>a</sup> Paula Escudeiro**

**Júri:**

Presidente:

Prof. Paulo Maio

Arguente:

Prof. Dr. Dirceu Teixeira

Porto, 15 de outubro de 2022



# Dedicatória

Dedico este trabalho a todos que estiveram presentes nesta etapa, por acreditarem nas minhas capacidades e pelo apoio dado ao longo de todo este percurso.



# Resumo

Ao longo dos anos o stress tem tido um papel negativo na vida dos colaboradores das organizações, podendo provocar doenças graves, tornando-se assim um problema na vida de muitas pessoas.

Nesta dissertação, será descrito o processo de recolha de informação, o desenho de uma solução, tal como o desenvolvimento e avaliação de um jogo sério, com o objetivo de auxiliar e aliviar o stress dos colaboradores das organizações. Este documento foi realizado ao longo do segundo ano letivo, referente à unidade curricular de Tese/Dissertação/Estágio do Mestrado de Engenharia Informática, na área de especialização Sistemas Gráficos e Multimédia do Instituto Superior de Engenharia do Porto.

Pretende-se que a solução desenvolvida forneça ao jogador, conhecimentos como aliviar o stress em ambiente cooperativo, indicando possíveis métodos para lidar com este ou métodos para prevenir. Para além disso, fornecer *biofeedback* ao utilizador.

**Palavras-chave:** Jogos Sérios, Unity, Informação, Stress, Ambiente Cooperativo, *Biofeedback*, Prevenção



# Abstract

Over the years, stress has played a negative role in the lives of employees in organizations, and can cause serious illness, thus becoming a problem in the lives of many people.

In this dissertation, the process of collecting information, the design of a solution, as well as the development and evaluation of a serious game will be described, with the aim of helping and relieving the stress of employees in organizations. This document was produced during the second academic year, referring to the curricular unit of Thesis/Dissertation/Internship of the master's in computer engineering, in specialization Graphic and Multimedia Systems of the Instituto Superior de Engenharia do Porto.

It is intended that the developed solution provides the player with knowledge on how to relieve stress in a cooperative environment, indicating possible methods to deal with it or methods to prevent it. In addition, provide biofeedback to the user.

**Keywords:** Serious Games, Unity, Information, Stress, Cooperative Environment, Biofeedback, Prevention



# Agradecimentos

O alcançar desta etapa não teria sido possível sem a colaboração, auxílio, carinho e dedicação por parte de várias pessoas ao longo de todo o percurso da minha formação. Por esta mesma razão, não quero deixar passar esta oportunidade para agradecer a todos aqueles que, direta ou indiretamente, contribuíram para o meu sucesso e a minha chegada até aqui.

Agradeço, em primeiro lugar, a todos os professores que me acompanharam, tanto no decorrer da licenciatura, como especialmente no mestrado.

Agradeço a minha orientadora, Doutora Paula Maria De Sá Oliveira Escudeiro, pelo apoio ao longo do desenvolvimento deste projeto.

Agradeço aos meus amigos, por me motivarem durante todo este percurso, auxiliarem no que fosse possível, pelas sinceras opiniões e por todo o apoio.

Por último agradeço especialmente aos meus pais pelo amor incondicional, pela dedicação, pelo incentivo a conquistar os meus objetivos, pela paciência, pois sem estes não teria alcançado esta etapa.



# Índice

<b>1</b>	<b>Introdução .....</b>	<b>21</b>
1.1	Contexto e problema .....	21
1.2	Objetivos .....	22
1.3	Conceitos .....	23
1.4	Fase de planeamento .....	23
1.4.1	Diagrama de Gantt .....	24
1.5	Estrutura do documento.....	24
<b>2</b>	<b>Estado da Arte .....</b>	<b>27</b>
2.1	Stress.....	27
2.1.1	Stress Positivo VS Stress Negativo.....	29
2.1.2	Métodos de medição .....	29
2.1.3	Aparelhos de medição .....	30
2.1.4	Perceived Stress Scale .....	32
2.2	Jogos e Jogos sérios.....	33
2.2.1	Importância de um jogo sério na Saúde.....	34
2.2.2	Exemplos de jogos no mercado atual.....	34
2.2.3	Dynamic Difficulty Adjustment .....	37
2.3	Relação entre o stress e os jogos - Competitividade.....	38
2.4	Motores de jogo .....	38
2.4.1	Unity .....	38
2.4.2	Unreal Engine .....	39
2.4.3	GameMaker.....	40
2.4.4	RPG Maker.....	40
2.5	Gestão de Projeto.....	41
<b>3</b>	<b>Análise de valor .....</b>	<b>43</b>
3.1	Modelo New Concept development .....	43
3.1.1	Analytic Hierarchy Process.....	47
3.2	Valor da solução.....	55
3.2.1	Proposta de valor.....	55
3.2.2	Quality Function Deployment.....	56
<b>4</b>	<b>Avaliação de Abordagens .....</b>	<b>61</b>
4.1	Stress e Abordagens.....	61
4.2	Motor de jogo selecionado .....	62
<b>5</b>	<b>Desenho .....</b>	<b>65</b>
5.1	Género de jogo escolhido .....	65

5.2	Dificuldade .....	65
5.3	Levantamento de requisitos .....	66
5.3.1	Requisitos funcionais .....	67
5.3.2	Requisitos não funcionais .....	69
5.4	Casos de Uso .....	70
5.5	Fluxograma.....	70
5.6	Base de dados .....	72
5.7	Storyboard .....	74
5.8	Paletes de cor a ser utilizada .....	77
<b>6</b>	<b>Construção da Solução .....</b>	<b>79</b>
6.1	Protótipo Desenvolvido.....	79
6.1.1	Logo do jogo .....	80
6.1.2	Menu principal .....	80
6.1.3	Definições .....	83
6.1.4	Questionário PSS .....	88
6.1.5	Resultado do Questionário .....	90
6.1.6	Sobre o jogo .....	91
6.1.7	Seleção de níveis .....	92
6.1.8	Carta e Coleção de cartas .....	93
6.1.9	Níveis .....	96
6.1.10	Como jogar .....	96
<b>7</b>	<b>Experimentação e avaliação.....</b>	<b>97</b>
7.1	Hipótese de investigação .....	97
7.2	Metodologia de avaliação .....	98
7.3	Indicadores e fontes de informação .....	98
7.4	Grandezas de avaliação .....	98
7.4.1	Quantitative Evaluation Framework.....	98
7.5	System Usability Scale.....	101
7.5.1	Questionário .....	101
7.5.2	Resultados de cada afirmação.....	102
7.5.3	Cálculo final .....	107
7.6	Apreciação dos Resultados Obtidos.....	108
<b>8</b>	<b>Conclusão .....</b>	<b>109</b>
8.1	Objetivos .....	109
8.1.1	Sucesso do protótipo.....	110
8.2	Trabalho Futuro .....	110
8.3	Apreciação final .....	111

# Lista de Figuras

Figura 1. Diagrama de GANTT .....	24
Figura 2. Muse 2 Headband (Muse, s.d.) .....	30
Figura 3. Leaf Urban (Leaf Urban, s.d.) .....	30
Figura 4. Spire Stone (Spire Health, s.d.).....	31
Figura 5. Versus Headset (Versus Headset, s.d.).....	31
Figura 6. Questionário PSS (Cohen) .....	32
Figura 7. Taxonomia dos Jogos Sérios (Ben & Peter, 2008).....	33
Figura 8. “Botanical Nerver” .....	34
Figura 9. Resultado do questionário 1 .....	36
Figura 10. Componentes mais stressantes num Vídeo Jogo.....	36
Figura 11. Jogos e Géneros menos stressantes .....	37
Figura 12. DDA (Zohaib, 2018) .....	37
Figura 13. Trello projeto.....	41
Figura 14. Processo de Inovação.....	43
Figura 15. O Modelo New Concept Development .....	44
Figura 16. Árvore hierárquica .....	47
Figura 17. Escala Fundamental. (Joost, Esther, & Bahareh, 2015).....	48
Figura 18. Diagrama com os resultados .....	54
Figura 19. Canvas da proposta de valor .....	56
Figura 20. Legenda dos Valores e símbolos .....	58
Figura 21. House of Quality.....	58
Figura 22. Características Unity (Eduardo, 2015).....	63
Figura 23. Diagrama de casos de Uso .....	70
Figura 24. Fluxograma.....	71
Figura 25. Esquema <i>json</i> parte 1. ....	72
Figura 26. Esquema <i>json</i> parte 2. ....	73
Figura 27. Esquema <i>json</i> parte 3. ....	73
Figura 28. Menu principal. ....	74
Figura 29. Menu definições.....	74
Figura 30. Sobre o jogo. ....	75
Figura 31. Como jogar. ....	75
Figura 32. Resultados obtidos do stress.....	75
Figura 33. Questionário PSS.....	76
Figura 34. Menu de níveis.....	76
Figura 35. Nível.....	77
Figura 36. Coleção de cartas. ....	77
Figura 37. Gradiente da paleta de cores.....	78
Figura 38. Paleta de cores.....	78
Figura 39. Logo do jogo.....	80
Figura 40. Interface Unity – Menu principal. ....	80

Figura 41. Código para mudar de cenário com <i>fader</i> .....	81
Figura 42. Botão para mudar de cenário.....	81
Figura 43. Código do Fader.....	81
Figura 44. Código do botão Resultados.....	82
Figura 45. Código Abrir Painel.....	82
Figura 46. Botão para mudar de painel.....	83
Figura 47. Código <i>Quit</i> .....	83
Figura 48. Menu definições – Unity .....	83
Figura 49. Código para guardar as preferências do jogador.....	84
Figura 50. Ficheiro multilang.....	85
Figura 51. Código de leitura dos idiomas.....	85
Figura 52. Código associado ao objeto.....	86
Figura 53. Código do <i>Dropdown</i> .....	86
Figura 54. Propriedades do Texto.....	87
Figura 55. Código para mudar o texto conforme idioma.....	87
Figura 56. Código do áudio.....	87
Figura 57. Preferências do <i>Slider</i> .....	88
Figura 58. Questionário PSS.....	88
Figura 59. Código do cálculo do PSS.....	89
Figura 60. Código que salva no csv.....	89
Figura 61. Ficheiro csv de resultados.....	90
Figura 62. Resultado do questionário.....	90
Figura 63. Código desenhar gráfico – Parte 1.....	90
Figura 64. Código desenhar gráfico – Parte 2.....	91
Figura 65. Sobre o jogo.....	91
Figura 66. Níveis bloqueados.....	92
Figura 67. Número de níveis disponíveis.....	92
Figura 68. Código de bloqueio dos níveis.....	93
Figura 69. Coleção de Cartas – Unity.....	93
Figura 70. Classe Carta.....	94
Figura 71. Código para ler o ficheiro – Coleção de cartas.....	94
Figura 72. Ficheiro coleção cartas.....	95
Figura 73. Atribuição dos objetos.....	95
Figura 74. Bloquear cartas ainda não adquiridas.....	95
Figura 75. Jogo de Puzzle.....	96
Figura 76. Código Adicionado ao asset.....	96
Figura 77. Como jogar – Manual.....	96
Figura 78. Resultado QEF .....	100
Figura 79. Escala de avaliação SUS.....	101
Figura 80. Cálculo executado por linha .....	107
Figura 81. Resultado dos Cálculos SUS.....	107
Figura 82. Escala de avaliação SUS.....	108

## Lista de gráficos

Gráfico 1. Resultado da afirmação “A ideia do jogo é original” .....	102
Gráfico 2. Resultado da afirmação “O jogo não é divertido de jogar, nem desafiante”. .....	103
Gráfico 3. Resultado da afirmação “Fiquei motivado para continuar a jogar” .....	103
Gráfico 4. Resultado da afirmação “Senti-me "preso" num nível, durante o jogo”. .....	104
Gráfico 5. Resultado da afirmação “O jogo não é intrusivo e é fácil de entender” .....	104
Gráfico 6. Resultado da afirmação relacionada com o som e os gráficos do jogo .....	105
Gráfico 7. Resultado da afirmação relacionada com o início do jogo.....	105
Gráfico 8. Resultado da afirmação sobre o controlo do jogo. ....	106
Gráfico 9. Resultado da afirmação relacionada com o conteúdo.....	106
Gráfico 10. Resultado a última afirmação.....	107



# Lista de Tabelas

Tabela 1. Escala de avaliação .....	32
Tabela 2. Matriz SWOT.....	46
Tabela 3. Matriz de comparação dos critérios.....	48
Tabela 4. Somatório das colunas anteriores.....	49
Tabela 5. Matriz normalizada dos critérios.....	49
Tabela 6. Peso dos critérios .....	49
Tabela 7. Índices aleatórios.....	50
Tabela 8. Matriz de comparação em relação ao Preço.....	51
Tabela 9. Somatório das colunas anteriores de preço.....	51
Tabela 10. Matriz normalizada das opções em relação com o Preço.....	51
Tabela 11. Vetor prioridade em relação ao Preço .....	51
Tabela 12. Matriz de comparação em relação à Saturação .....	52
Tabela 13. Vetor prioridade em relação à Saturação .....	52
Tabela 14. Matriz de comparação em relação aos Gráficos .....	52
Tabela 15. Vetor prioridade em relação ao Gráficos .....	52
Tabela 16. Matriz de comparação em relação à Motivação.....	53
Tabela 17. Vetor prioridade em relação à Motivação .....	53
Tabela 18. Matriz de comparação em relação ao Material de Apoio.....	53
Tabela 19. Vetor prioridade em relação ao Material de Apoio .....	54
Tabela 20. Pontuação por tempo.....	66
Tabela 21. Relação número de peças e nível.....	66
Tabela 22. Requisito Funcional Número 1.....	67
Tabela 23. Requisito Funcional Número 2.....	68
Tabela 24. Requisito Funcional Número 3.....	68
Tabela 25. Requisito Funcional Número 4.....	68
Tabela 26. Requisito Funcional Número 5.....	68
Tabela 27. Requisito Funcional Número 6.....	68
Tabela 28. Requisito Funcional Número 7.....	68
Tabela 29. Requisito Funcional Número 8.....	68
Tabela 30. Requisito Funcional Número 9.....	69
Tabela 31. Requisito Funcional Número 10.....	69
Tabela 32. Requisito Funcional Número 11 .....	69
Tabela 33. Tabela de Hipóteses .....	97
Tabela 34. Grandezas de avaliação.....	98
Tabela 35. QEF Dimensões e Fatores.....	99
Tabela 36. Perguntas do Questionário.....	101
Tabela 37. Número de vezes ocorrido uma avaliação. ....	102



# Acrónimos e Símbolos

## Lista de Acrónimos

<b>AHP</b>	Analytic Hierarchy Process
<b>ASN</b>	Estado do sistema nervoso Autónomo
<b>BD</b>	Base de dados
<b>COVID-19</b>	SARS-CoV-2
<b>DDA</b>	Dynamic Difficulty Adjustment
<b>FEI</b>	The front end of innovation
<b>GILT</b>	Graphics Interaction Learning Technologies
<b>HOQ</b>	House Of Quality
<b>HRV</b>	Heart Rate Variability analysis
<b>ISEP</b>	Instituto Superior de Engenharia do Porto
<b>NCD</b>	New Concept Development
<b>NPD</b>	New Product Development
<b>NPPD</b>	New Product and process development
<b>ONG</b>	Organizações Não-Governamentais
<b>PSS</b>	Perceived Stress Scale
<b>QEF</b>	Quantitative Evaluation Framework
<b>RF</b>	Requisitos Funcionais
<b>SUS</b>	System Usability Scale
<b>TMDEI</b>	Unidade curricular de Tese/Dissertação/Estágio



# 1 Introdução

Este documento foi realizado ao longo do segundo ano letivo, referente à unidade curricular de Tese/Dissertação/Estágio (TMDEI) do Mestrado de Engenharia Informática, na área de especialização Sistemas Gráficos e Multimédia do Instituto Superior de Engenharia do Porto (ISEP).

O projeto foi proposto pelo grupo de *Investigação Games Interaction and Learning Technologies* (GILT), do ISEP, sendo o tema escolhido o desenvolvimento de um jogo sério de modo a auxiliar na prevenção do stress em espaços Profissionais.

Neste primeiro capítulo é realizado um enquadramento e contextualização do problema em questão, de seguida uma breve descrição dos objetivos pretendidos, apresentação do planeamento delineado para a conclusão do projeto, bem como a estrutura do presente documento.

## 1.1 Contexto e problema

Ao longo dos restantes anos letivos, a autora do presente documento, sendo estudante no ISEP, na área de especialização sistemas gráficos e multimédia, adquiriu conhecimentos sobre a importância e crescimento dos jogos sérios na sociedade atual. Sendo este um tema com alargada informação para explorar. Foi executada a escolha de desenvolver um jogo sério e documentar todo esse processo, com o objetivo, no final, de entregar uma dissertação.

Para além dessa escolha, sendo também, colaboradora numa empresa, veio a enfrentar algumas dificuldades relacionadas com o problema que o jogo sério pretende abranger, este será referido no próximo parágrafo. No processo de escolha do tema, para desenvolvimento da dissertação, esta sensibilizou-se com o tema apresentado, pretendendo diminuir ou apoiar uma possível resolução para o problema.

Sendo assim, foi selecionado o tema desenvolvimento de um jogo sério para a Prevenção do Stress em Espaços Profissionais.

Em ambientes corporativos, devido à necessidade de esforço maior pelos colaboradores de modo a “fazer mais com menos”, estes estão sujeitos a um ambiente que poderá ser caracterizado em três aspetos, estes deverão ser considerados para uma melhor análise do problema:

- *Never offline*: pela necessidade de os colaboradores estarem sempre contactáveis.
- Pressão para obter resultados: pela necessidade de entrega dos projetos de acordo com o plano e cumprir os prazos estabelecidos, os colaboradores sofrem uma pressão para obter resultados, estes níveis de pressão irão adicionar stress.
- Longas horas de trabalho: para além das horas de trabalho, muitos colaboradores prolongam este horário, acabando por trabalhar em “dois turnos” (Soares, 2018).

Estes três fatores poderão levar a dois conceitos muito conhecidos no mundo cooperativo: **stress** e **fadiga**. Sendo assim o primeiro conceito é um problema muito comum no ambiente de negócios que afeta muitos trabalhadores e pode até causar sintomas físicos, como dores de cabeça, dor de estômago, aumento da pressão arterial, problemas para adormecer ou até insónias. No caso se este for mais severo poderá levar à depressão.

Em suma, o stress é um problema grave, presente na vida dos colaboradores. Este veio a piorar com o aparecimento do COVID-19, aquando dos ambientes cooperativos tiveram de optar pela metodologia “*Home Office*”, ou trabalho remoto. Mesmo antes do aparecimento do COVID-19, o inquérito realizado pela *Nova Information Management School*, da Universidade Nova de Lisboa indicava que trabalhar remotamente é mais stressante. Este estudo indica que 64%, sentiram um aumento dos níveis de stress em período laboral, tal como este problema é presente e veio a aumentar em 90% dos jovens entre os 18 e os 24 anos (Staff, 2017).

## 1.2 Objetivos

O objetivo central será desenvolver um jogo sério, de modo a sensibilizar os colaboradores das organizações empresariais que o stress é um problema. Este poderá prejudicar futuramente, tal como referido na secção 1.1, além disso, pretende-se desenvolver um jogo que auxilie para a prevenção do stress. Sendo assim, foram gerados os seguintes objetivos:

- Identificar e analisar o problema (cf. secção 1.1);
- Identificar alternativas existentes no mercado (cf. secção 2.2.2);
- Desenvolver um jogo sério que auxilie no combate e na prevenção do stress em ambientes profissionais;
- Experimentação e avaliação do projeto desenvolvido (cf. secção 0);

## 1.3 Conceitos

Previamente é necessária uma breve apresentação de dois conceitos importantes para melhor compreensão do objetivo principal deste documento. Os conceitos são o que consiste um jogo e um jogo sério, tal como o que se trata o stress.

Um **jogo** é uma atividade em que podem ser executadas individualmente, tal como com mais participantes. O principal objetivo deste é o entretenimento. No caso de um **jogo sério**, este perde como o objetivo principal o entretenimento, e passa a ser o fornecimento de informação, de modo a assumir um papel educativo, ou informativo.

O **stress**, é a resposta do corpo humano a fatores externos, como internos que afetam o equilíbrio de um indivíduo, tanto psicologicamente como fisicamente.

## 1.4 Fase de planeamento

Para melhor planeamento de tarefas ao longo da realização do projeto, foi decidido dividir este percurso em 5 fases, sendo estas: pesquisa e análise, desenho da solução, desenvolvimento, experimentação e avaliação, e documentação.

A fase **Documentação**, representa a tarefa de escrita da dissertação, esta será realizada ao longo das fases restantes. Esta finalizou-se em outubro.

Na primeira fase **Pesquisa e Análise**, nesta foi executada pesquisas sobre o problema que se pretende prevenir e abranger, tais como mecanismos de medições deste e aparelhos para este efeito. Foram encontradas alternativas do jogo no mercado atual, executada uma identificação sobre as principais tecnologias, para a próxima fase. Por último foi executado uma análise de valor. A fase iniciou-se em novembro e finalizou-se em fevereiro.

Na segunda fase, **Desenho da Solução**, foram identificadas as soluções que se pretende aplicar no projeto, por exemplo, levantamento de requisitos, desenvolvimento dos diagramas necessários para melhor compreensão do que será desenvolvido na próxima fase. Esta fase teve início em fevereiro e finalizou-se em maio.

Na fase **Desenvolvimento**, será a fase dedicada ao desenvolvimento do projeto, nas quais, serão implementadas as soluções definidas durante a fase de anterior. Iniciou-se em maio e finalizou-se em outubro.

Na quarta fase, **Experimentação e Avaliação**, será aplicada a metodologia escolhida, de modo a avaliar a solução implementada, de modo a verificar possíveis melhorias e problemas. Esta última fase terá início após a fase anterior tenha sido terminada.

### 1.4.1 Diagrama de Gantt

Para uma melhor visualização do planeamento, foi desenvolvido um **Diagrama de Gantt**. Este foi criado por Henry Gantt em 1917, sendo uma ferramenta que permite elaborar e controlar o cronograma de um projeto.

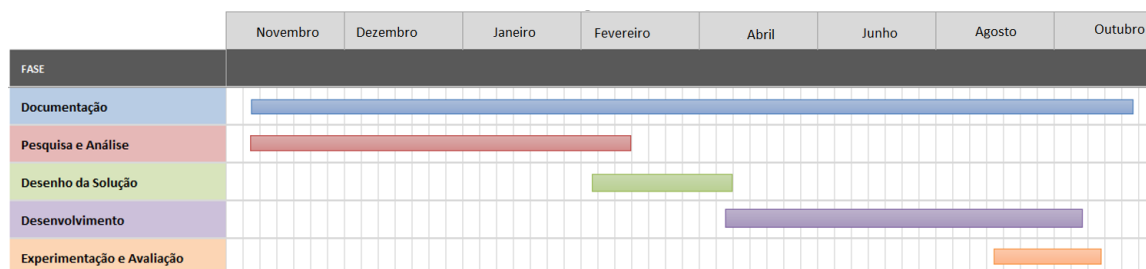


Figura 1. Diagrama de GANTT

## 1.5 Estrutura do documento

O presente documento irá ser dividido em oito capítulos, que irão constituir o corpo principal da dissertação.

No primeiro capítulo, Introdução, é executada uma introdução breve ao problema a que se trata o desenvolvimento da dissertação. Descrito quais os objetivos que se pretende cumprir com o desenvolvimento do jogo sério, e uma descrição do que será executado ao longo do projeto.

No segundo capítulo, Estado da Arte, é realizada uma recolha de informação sobre o problema, mecanismos de medição e aparelhos para tal. É também, feita uma pesquisa sobre os jogos e jogos sérios e alternativas no mercado, tal como apresentação de algumas tecnologias.

O terceiro capítulo,

Análise de valor, é apresentada o valor do projeto, tal como, da seleção de oportunidades mais apropriadas para este.

No quarto capítulo, Avaliação de Abordagens, é feita uma análise de como abordar o problema a resolver, tal como uma avaliação ao motor de jogo escolhido no capítulo anterior.

No quinto capítulo, Desenho, é executado o desenho de uma possível solução a ser implementada, sendo neste, onde será feita a descrição de tipo de jogo escolhido, e o levantamento de requisitos.

No sexto capítulo, Construção da Solução, será apresentada a solução desenvolvida, após a recolha de informação executada no segundo capítulo e do desenho da solução.

No sétimo capítulo deste documento, Experimentação e avaliação, é apresentado o método de avaliação que será executado após e durante o desenvolvimento deste projeto.

Por último, Conclusão, serão apresentadas as conclusões adquiridas no final do desenvolvimento deste projeto, juntamente com sugestões para futuros desenvolvimentos, que poderão ser realizados.



## 2 Estado da Arte

Neste capítulo será apresentado o resultado da identificação e análise executada sobre o problema, sobre os meios de medição e os aparelhos que são utilizados para as medições. Para além do descrito anteriormente, serão apresentadas as alternativas do mercado, até à atualidade.

### 2.1 Stress

De acordo como livro *Reload* (Soares, 2018, p. 27), o termo stress foi utilizado pela primeira vez, de um ponto de vista científico nos anos 30, pelo endocrinologista *Hans Selye*. O cientista “colocou animais sob diferentes estímulos nocivos e reparou que todos eles, independentemente do agente usado”, reponderam da mesma maneira, acabou por designar esta resposta como Síndrome Geral de adaptação. Este contem as seguintes três fases:

1. Fase de alarme, corresponde identificação da ameaça ou agente de Stress e a preparação para a reação.
2. Fase de resistência, o corpo tenta adaptar-se e lidar com o stress que esta a ser imposto.
3. Fase de exaustão, a capacidade de resposta se esgota, e o corpo acaba por não conseguir resistir ao agente de perturbação do equilíbrio.

Pode-se concluir que o stress é a reação do organismo de um ser vivo a um estímulo/agente que perturba o equilíbrio físico e/ou psicológico. Este pode ser interno ou externo. Segue-se uma breve descrição de cada um com exemplos:

- Interno: Enquadra-se as doenças, por exemplo, um vírus, ou até mesmo alguns procedimentos médicos, como raio-X, colheita de sangue, entre outros.
- Externo: Neste grupo encontram-se fatores como quando o coração tem um batimento acelerado, quando a digestão para de funcionar. Estes fatores podem ser causados por ações, por exemplo, antes de uma apresentação, ou a espera de um resultado médico.

Atualmente o agente mais presente e predominante, que possui um grande impacto no dia a dia de uma pessoa é o Tempo. Numa conversa seja pessoal ou profissional, pode-se ouvir várias vezes a palavra tempo surgir juntamente com a palavra stress, expressões como “Não tenho tempo para entregar o relatório, não tenho tempo para reunir com a equipa, não tenho tempo para estar com amigos” entre outras, são parte dessa conversa.

Segundo a Agência Europeia para a Segurança e Saúde no Trabalho (Trabalho A. E., 2014), em 2014, foram estimados que os custos do stress e da saúde mental, na Europa, seja mais que 250 mil milhões de euros. Parte dos custos referidos anteriormente, estará relacionado com o “turnover” destes, ou seja, diminuição da produtividade, custos diretos relacionados com despesas médicas, legais, seguros e indemnizações. Pode-se concluir que o stress tem um impacto prejudicial nos colaboradores de uma organização tal como é uma entidade com um grande impacto na produtividade, saúde e bem-estar destes.

O stress é normalmente dividido em dois tipos:

- Agudo: resposta imediata a algo que perturbe o indivíduo naquele momento, independentemente da origem.
- Crónico: quando existe um prolongamento de stress agudo este passa a ser crónico.

Quando o stress é a longo termo ou crónico, irá fazer que o organismo do indivíduo reaja. Esta reação poderá originar doenças e disfunções, por exemplo, a depressão, excesso de peso e obesidade, acidente cardiovascular, problemas de memória, diminuição da imunidade e disfunção do sono (Soares, 2018, p. 31).

Alguns exemplos de condições em ambiente empresarial (Trabalho A. E., s.d.) que podem levar o colaborador a estado de stress são:

- Excesso de carga de trabalho;
- A falta de organização por parte do funcionário ou por parte dos superiores;
- Falta de clareza nas descrições de tarefas;
- Comunicação ineficaz;
- Falta de apoio da parte de chefias e colegas;
- Baixa autoestima;

### 2.1.1 Stress Positivo VS Stress Negativo

Para além das informações ditas anteriormente, o stress, sendo a reação do corpo humano a um fator externo, este poderá ser diferenciado, como positivo ou negativo. (Hepburn, 2019)

O stress negativo, é normalmente causado por situações ou cenários, em que faça sentir uma sobrecarga, ou até mesmo pressão, nas se tem pouco controlo, ou até mesmo nenhum controlo sobre a situação. Alguns exemplos de situações que poderão vir a causar stress são:

- Problemas financeiros.
- Problemas com a justiça.
- Problemas de relacionamentos.
- Lesões ou doenças.
- Insegurança no trabalho.
- Horas excessivas e sobrecarga no trabalho.

O stress positivo, normalmente, pode ser causado por cenários que trazem mudanças no dia a dia do ser humano, mesmo sendo desafiador. Alguns exemplos:

- Começar um novo emprego.
- Mudar de cidade.
- A notícia de ter um filho.
- Início de um novo projeto.

### 2.1.2 Métodos de medição

Foi executada uma pesquisa de mecanismos ou métodos utilizados para a medição do stress, para mais tarde optar e implementar no jogo.

Estes mecanismos são através de *biofeedback*. O *biofeedback*, de acordo com Kendra Cherry (Cherry, 2021), é uma técnica que envolve o uso de *feedback* visual ou auditivo para ensinar as pessoas a reconhecer sinais e sintomas físicos, neste caso este será utilizado para recolher informações sobre os níveis de stress de uma pessoa. Alguns exemplos de *biofeedback* são o aumento da frequência cardíaca, temperatura corporal e tensão muscular.

Existem várias maneiras, sendo que só se irá descrever as mais conhecidas (How To Measure Stress, s.d.), nas quais:

- *Heart Rate Variability* (HRV) – Baseia-se na mudança da frequência cardíaca durante um período.
- *Neurofeedback* – Também conhecido como EEG *biofeedback*, é uma voltagem medida a partir do couro cabeludo, medindo as ondas cerebrais.
- Avaliação da respiração – Uma maneira comum do corpo humano representar que esta a passar por níveis de stress, é através da respiração, sendo que esta fica mais excessiva.

### 2.1.3 Aparelhos de medição

A maioria dos dispositivos que medem o stress, utilizam *biofeedback*, seguem-se alguns exemplos (Eatough, 2021):

#### 1. *Muse Headbands*

Dispositivo que utiliza sensores de EEG para medir a atividade cerebral do utilizador, conecta-se através de uma aplicação. Esta possui meditações guiadas e exercícios de respiração. O preço deste dispositivo varia entre 270 e 490 euros.

É uma solução de alta tecnologia que deteta altos níveis de stress e mapeia a atividade cerebral, caracterizando em grupos, nas quais, ativo, neutro e calmo.

Design elegante, adequado a todas as idades e níveis de atividade. No entanto, pela opinião de alguns utilizadores, este só é eficaz quando se encaixa bem na testa, sendo que com pequenos movimentos da cabeça poderão afetar os resultados.



Figura 2. Muse 2 Headband (Muse, s.d.)

#### 2. *Stress Tracker Leaf Urban* da Bellabeat

Trata-se de um dispositivo, com um design prático e atraente, pois se parece com uma pulseira, para usar diariamente. O Preço ronda cerca de 87 euros.

Este oferece uma variedade de métricas de rastreamento de stress básico, mas informativas. Com o tempo, ele pode armazenar uma variedade de dados de saúde e bem-estar dos utilizadores, numa única aplicação personalizada.



Figura 3. Leaf Urban (Leaf Urban, s.d.)

### 3. Apple Watch / Smart Watch

A série *Apple Watch* já é conhecida pelos recursos de saúde e bem-estar, mas desde a Série 6, foi adicionado a esses recursos uma variedade de funções de monitorização do stress e sono.

Para além dos *Apple Watches*, existe uma grande variedade de *smart watches*, que executam a mesma medição. Estes são os *wearables* mais conhecidos e utilizados atualmente na sociedade.

Os preços variam entre os 200 e 800 euros.

### 4. Health tag Spire Stone

Aparelho portátil, em que o utilizador pode prender na camisola, este rastreia a respiração do utilizador durante 24 horas, 7 dias por semana, através do movimento do tronco e do peito, enviando alertas para o smartphone quando a taxa de respiração aumenta. A Spire Stone não tem ecrã, ou seja, é necessário um smartphone ou tablet para poder aceder à aplicação. Atualmente após pesquisa não foi encontrado valor disponível.



Figura 4. Spire Stone (Spire Health, s.d.)

### 5. Headset da Versus

São *headsets*, ou auscultadores, da marca Versus, estes avaliam os níveis de stress através de sensores EEG. Conectam-se por aplicação. Não é um dispositivo engenhoso para a utilização diária.

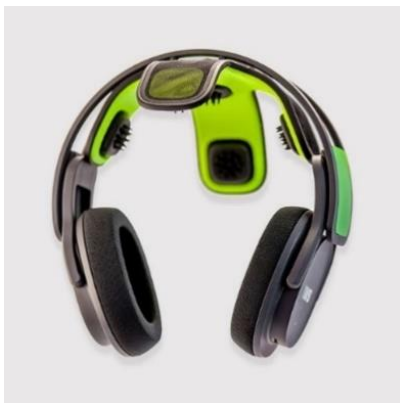


Figura 5. Versus Headset (Versus Headset, s.d.)

### 2.1.4 Perceived Stress Scale

O Perceived Stress Scale (PSS), criado por Cohen Et Al, em 1983 (TEST YOUR STRESS, s.d.) é um método para medição do stress, sendo que este não utiliza *biofeedback*, tornando-se outra alternativa possível para implementar no jogo.

Este é um questionário, composto por dez perguntas (Figura 6. Questionário PSS Figura 6) em que o utilizador deverá avaliar, de acordo com uma escala fornecida, ver

Tabela 1. Após isso, o cálculo é a soma de todos os valores, tendo em consideração que nas questões número quatro, cinco, sete e oito, os valores invertem, por exemplo, o valor zero passa a ser igual a quatro (0 = 4), e o valor quatro passa a valer zero (4=0).

	0 = Never	1 = Almost Never	2 = Sometimes	3 = Fairly Often	4 = Very Often
1. In the last month, how often have you been upset because of something that happened unexpectedly? .....	0	1	2	3	4
2. In the last month, how often have you felt that you were unable to control the important things in your life? .....	0	1	2	3	4
3. In the last month, how often have you felt nervous and "stressed"? .....	0	1	2	3	4
4. In the last month, how often have you felt confident about your ability to handle your personal problems? .....	0	1	2	3	4
5. In the last month, how often have you felt that things were going your way?.....	0	1	2	3	4
6. In the last month, how often have you found that you could not cope with all the things that you had to do? .....	0	1	2	3	4
7. In the last month, how often have you been able to control irritations in your life?.....	0	1	2	3	4
8. In the last month, how often have you felt that you were on top of things?..	0	1	2	3	4
9. In the last month, how often have you been angered because of things that were outside of your control?.....	0	1	2	3	4
10. In the last month, how often have you felt difficulties were piling up so high that you could not overcome them? .....	0	1	2	3	4

Figura 6. Questionário PSS (Cohen)

Tabela 1. Escala de avaliação

0	Nunca
1	Quase nunca
2	Às vezes
3	Com bastante frequência
4	Muitas vezes

O resultado irá variar entre zero e quarenta, sendo este o valor máximo do questionário. No caso de o resultado ser entre zero e treze, é considerado que o individuo possui um stress baixo, entre catorze e vinte-seis, é considerado stress moderado, no caso de vinte e sete até quarenta, é considerado stress elevado.

## 2.2 Jogos e Jogos sérios

O conceito de jogo (Conceito de jogo, 2020), refere se a uma atividade recreativa em que participa um ou mais participantes, ou seja, *singleplayer* ou *Multiplayer*. A primeira função consiste em proporcionar entretenimento, lazer e diversão, embora também possa assumir um papel educativo.

De acordo com David R. Michael e Sandra L. Chen (2006 jan.), o conceito de jogo sério consiste num jogo em que o objetivo principal não seja o entretenimento. Consiste num desafio mental ou físico que usa entretenimento para promover treinos, políticas públicas, consciência social, saúde pública ou comunicação e marketing (Mike Zyda).

No entanto, na *Serious Games Summit* em 2008, Ben Sayer e Peter Smith (Ben & Peter, 2008) foi apresentada uma tipologia, de forma a dividir os jogos, não só, por áreas de atuação, mas também por enquadrá-las num determinado conteúdo. Sendo assim, a tipologia é dividida nas seguintes áreas, o Governo e as organizações não governamentais (ONGs), Defesa, Saúde, Marketing e comunicação, educação, ambiente cooperativo e por fim indústria. Em termos de conteúdos, estes podem dividir-se em Saúde, *Advergames*, Jogos para Treino, Jogos para Educação, Jogos para a Ciência e investigação, Produção e Jogos para o trabalho.

	Games for Health	Advergames	Games for Training	Games for Education	Games for Science and Research	Production	Games as Work
Government & NGO	Public Health Education & Mass Casualty Response	Political Games	Employee Training	Inform Public	Data Collection / Planning	Strategic & Policy Planning	Public Diplomacy, Opinion Research
Defense	Rehabilitation & Wellness	Recruitment & Propaganda	Soldier/Support Training	School House Education	Wargames / planning	War planning & weapons research	Command & Control
Healthcare	Cybertherapy / Exergaming	Public Health Policy & Social Awareness Campaigns	Training Games for Health Professionals	Games for Patient Education and Disease Management	Visualization & Epidemiology	Biotech manufacturing & design	Public Health Response Planning & Logistics
Marketing & Communications	Advertising Treatment	Advertising, marketing with games, product placement	Product Use	Product Information	Opinion Research	Machinima	Opinion Research
Education	Inform about diseases/risks	Social Issue Games	Train teachers / Train workforce skills	Learning	Computer Science & Recruitment	P2P Learning Constructivism Documentary?	Teaching Distance Learning
Corporate	Employee Health Information & Wellness	Customer Education & Awareness	Employee Training	Continuing Education & Certification	Advertising / visualization	Strategic Planning	Command & Control
Industry	Occupational Safety	Sales & Recruitment	Employee Training	Workforce Education	Process Optimization Simulation	Nano/Bio-tech Design	Command & Control

Figura 7. Taxonomia dos Jogos Sérios (Ben & Peter, 2008)

Ao enquadrar o jogo que é pretendido desenvolver na taxonomia dos jogos sérios, sendo o objetivo deste é prevenir o stress, encaixa-se na coluna de Games for *Health*, e sendo relacionado com um ambiente cooperativo, pertence à linha *Corporate*, neste caso o jogo, irá pertencer ao grupo de “*Employee health information e wellness*”. É pretendido com este fornecer conhecimento ao colaborador sobre o stress, tal como investir no bem-estar deste.

### 2.2.1 Importância de um jogo sério na Saúde

Considerando o crescimento da indústria tecnológica em específico os vídeos jogos, faz cada vez mais sentido, questionar sobre a utilização de videojogos na saúde. Atualmente, a medicina já percebeu, que utilizando os videojogos pode ter um grande benefício nesta, entre ajudar a preparar cirurgias, ajudar a promover o bem-estar, entre muitos mais.

Sendo assim, cada vez mais os jogos sérios se tornam importantes para a saúde, sendo outro mecanismo para lidar com os pacientes e praticar medicina.

### 2.2.2 Exemplos de jogos no mercado atual

De seguida serão apresentados exemplos de jogos sérios e jogos utilizados de modo a aliviar o stress, nas quais, *Bottanical Nerver*, *Decisions Under Pressure*, entre outros.

#### 2.2.2.1 “Bottanical Nerver”

Jogo sério para telemóvel, desenvolvido por *Hussein Al Osman*, *Haiwei Dong* e *Abdulmotaieb El Saddik* (IEEE, 2016). Possui como objetivo, gerir o nível de stress utilizando o HRV para calcular o batimento cardíaco do jogador.

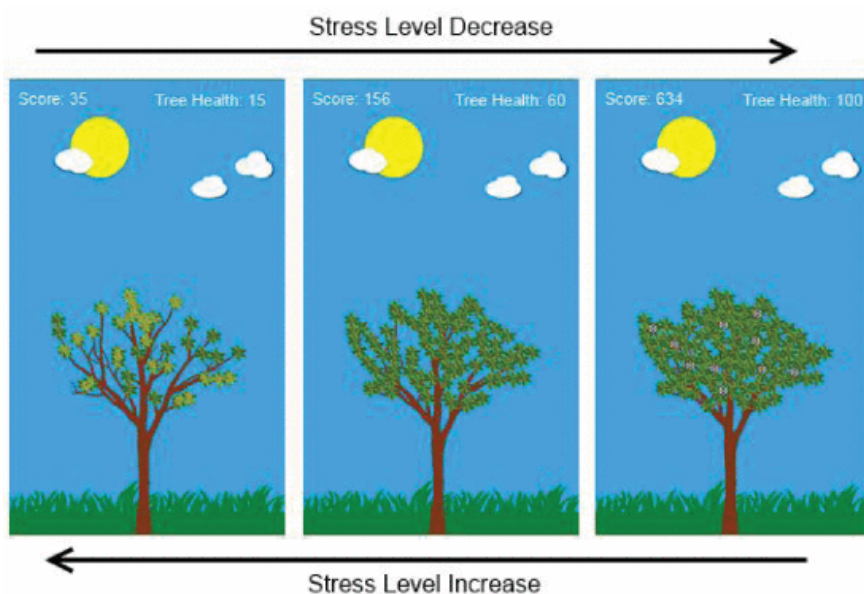


Figura 8. “Bottanical Nerver”

A única personagem do jogo é uma árvore, esta representa o estado do sistema nervoso Autônomo (ASN) do jogador. Sendo assim sempre que o jogador estiver sob pressão a saúde da árvore irá deteriorar, representada por uma animação das folhas da árvore amarelando e eventualmente irão cair. Quando o jogador começa a relaxar, as folhas vão começar a crescer, as existentes irão ficar verdes e eventualmente a árvore irá começar a florescer. A pontuação deste jogo é exibida na tela principal e esta indica as melhorias do funcionamento da árvore ao longo do tempo como um resumo do desempenho em geral.

#### 2.2.2.2 “Decisions Under Pressure”

Os investigadores da *Radboud*, *Karin Roelofs*, *Isabela Granic* e *Floris Klumpers* (*Serious games for police officers under stress*, 2017), iriam desenvolver um jogo sério, com o objetivo de treinar os policias, num ambiente de realidade virtual interativo, a responder da melhor forma em situações de stress. Estas poderão ser um tiroteio, violência doméstica, confronto com *hooligans*, e uma prisão. Durante este treino, a frequência cardíaca e as ondas teta<sup>1</sup> seriam medidos.

O objetivo geral do jogo é fornecer informações ao vivo (*Biofeedback*) aos polícias, de modo a estes aprenderem a colocar o corpo num estado ideal para a tomada de decisões, mesmo estando sob stress.

#### 2.2.2.3 Seis videojogos para combater o stress

De acordo com o artigo publicado na Time Out por Luís Filipe Rodrigues (Rodrigues, 2020), segue-se os seis melhores jogos para combater o stress.

1. Animal Crossing: New Horizons.
2. Stardew Valley.
3. The Legend of Zelda: Breath of the wild.
4. EA Sports UFC 4.
5. Street Fighter V.
6. Streets of Rage 4.

#### 2.2.2.4 Confy Sacks - Most Stressful Video Games

Neste estudo, que será descrito os resultados, executado em 2020, pela Confy Sacks (Confy Sacks, 2020), foram questionadas mais de 1.000 de jogadores sobre um dos efeitos dos jogos, neste caso o stress, apesar de não ser uma referência científica, foi adicionado ao documento como um facto curioso.

---

<sup>1</sup> Tipo de ondas cerebrais, que indica o estado de relaxamento e meditação. (Healing, s.d.)

Através da análise, fornecida na Figura 9, consegue-se perceber que no final dos questionários o jogo considerado por mais stressante, é o Call of Duty: Modern Warfare, tal como se consegue perceber qual o tipo de jogo considerados como mais stressantes, nas quais: “*First-Person Shooter*”.

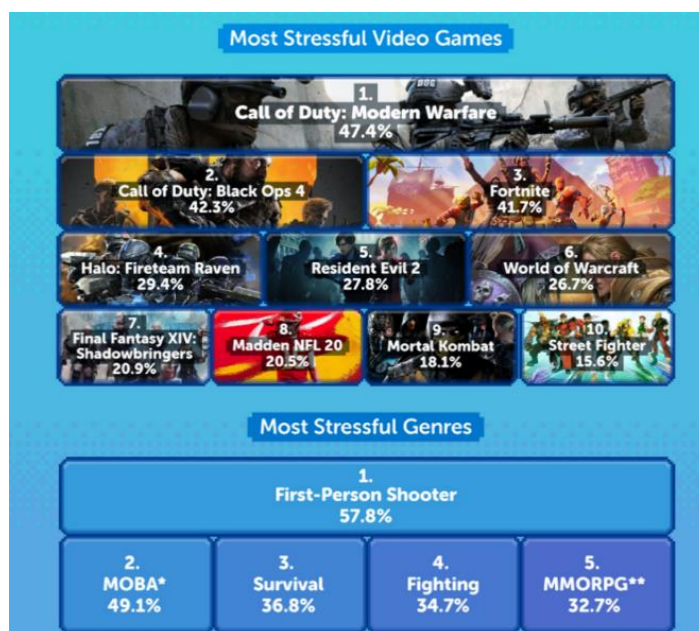


Figura 9. Resultado do questionário 1

Sendo que o componente mais stressante, de acordo com os sujeitos de teste, é a componente de dificuldade, ou seja, a medida que a dificuldade aumenta os jogadores sentem-se mais stressados, tal como o **fator de competitividade**.



Figura 10. Componentes mais stressantes num Vídeo Jogo

Para além dos resultados descritos anteriormente, foram apresentados quais os jogos que mais aliviam o stress, estando em primeiro lugar o *The Sims*, como se pode verificar na Figura 11. Tal como os géneros de jogos considerados menos stressantes, nas quais 50,6% considerou os jogos do género puzzle, em primeiro lugar, depois *role-playing* com 50,3% e em terceiro lugar os jogos do género simulação.



Figura 11. Jogos e Géneros menos stressantes

### 2.2.3 Dynamic Difficulty Adjustment

O *Dynamic Difficulty Adjustment* (DDA), é um método utilizado nos jogos que modifica automaticamente a dificuldade de um jogo dependendo da habilidade do jogador. Deste modo, pretende-se evitar que um jogador fique entediado, caso o jogo seja muito fácil, ou frustrado, o que pode levar ao aumento de stress devido ao jogo ser demasiado difícil. (EBUN-AMU, 2021)

O objetivo do DDA é possibilitar um estado de equilíbrio entre a dificuldade do jogo e a experiência do jogador (Figura 12).

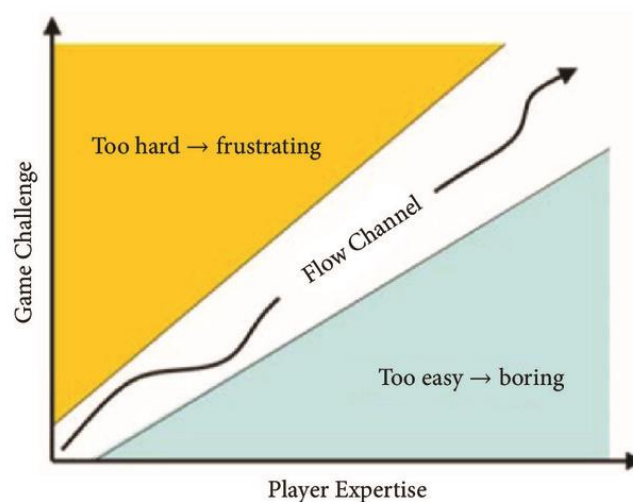


Figura 12. DDA (Zohaib, 2018)

## 2.3 Relação entre o stress e os jogos - Competitividade

A competitividade (Junior, 2002), pode ser dividida em quatro fases, nas quais se pode aplicar, tanto num desporto competitivo, como até mesmo num vídeo jogo. As fases serão descritas num ponto de vista de um jogador de vídeo jogos online, tendo como exemplo, a experiência da autora no jogo *League of Legends*, um jogo do género MOBA, em inglês *multiplayer online battle arena*, criado pela *Riot Games*.

- A primeira fase é o **confronto**, não existe competitividade sem confronto, neste caso esta é a primeira fase, nas quais onde o jogador, ou equipa, confronta inicialmente os oponentes.
- A segunda fase é a **demonstração**, onde o jogador irá pôr em prática as suas capacidades de jogar.
- De seguida, vem a **comparação**, no caso deste vídeo jogo, a comparação passa por ser quem consegue, aniquilar mais inimigos, de modo a conseguir ganhar mais pontos.
- E por último a **avaliação**, esta trata-se do resultado do jogo, se o jogador obteve a vitória ou não.

Com todas estas fases, o jogador, mesmo preparado, sofre pressão para “tentar ser o melhor”, este pensamento, leva a causa de stress excessivo, durante um jogo. Em suma, o fator competitividade de um jogo, poderá levar ao aumento do stress, sendo assim a importância da saliência deste tema.

## 2.4 Motores de jogo

Um Motor de jogo (em inglês, *Game Engine*) é um conjunto de ferramentas, que servem para facilitar no desenvolvimento de um jogo eletrónico e outros tipos de aplicações. Os motores de jogo possuem, em geral, um motor gráfico e outro de física. O primeiro serve para renderizar os gráficos em 2D/3D, enquanto o outro trabalha na simulação de efeitos físicos, como a deteção de colisão, além de suporte a animação e sons. Com a utilização deste, não é necessário codificar um jogo desde o zero, o que torna possível desenvolver um jogo, mesmo se o criador não saiba programar.

### 2.4.1 Unity

*Unity* é um motor de jogo criado pela *Unity Technologies*. Tornou-se numas das *engines* mais populares entre desenvolvedores independentes após os criadores terem lançado uma edição gratuita, em 2009. Este permite criar jogos para smartphones, tablets, consolas, browsers e Computadores Pessoais, funciona tanto no *Windows* quanto no *Mac OS* (Fidelis, 2018).

Algumas vantagens da utilização do *Unity* são:

- Este motor de jogo possui gráficos profissionais.
- A interface fácil de utilizar.
- Suporta *C#* e *JavaScript*.
- Fornece scripts prontos.
- Comunidade e tutoriais em abundância na internet.
- Facilidade na criação de jogos para android.
- Possibilidade de exportar para várias plataformas.
- Possibilidade de criar jogos 2D e 3D e multiplayer.
- Suporta realidade aumentada.

No caso de desvantagens são:

- Não permite modelação.
- Para obter todos os recursos, é necessário usar a versão paga.
- Requer conhecimentos de programação.
- Para vender os jogos é necessário pagar uma percentagem para a Unity, após um determinado número de vendas.

Alguns exemplos de jogos criados no Unity são o Rick and Morty: Virtual Rick-ality e o Hearthstone.

## 2.4.2 Unreal Engine

*Unreal Engine* é um motor de jogo, desenvolvida pela *Epic Games*. Foi utilizada pela primeira vez para produzir um jogo de tiros. O núcleo desta *engine* é escrito na linguagem de programação C++, o que possibilita uma boa portabilidade para diversas consolas e sistemas operacionais. Alguns exemplos de jogos criados nesta *engine*, são *Mortal Kombat X*, *Batman*, *Borderlands*, *Tekken 7* e *BioShock 2* (Fidelis, 2018).

Algumas vantagens são:

- Interface é intuitiva.
- É grátis.
- Cria jogos profissionais.
- Permite fazer animações
- Gráficos de última geração.
- Exporta para várias plataformas.
- Cria jogos sem saber programação.
- Permite desenvolver jogos com realidade aumentada.

No caso de desvantagens são:

- Para vender jogos tem de se pagar x por cento para a ferramenta.
- Não suporta outras linguagens de programação.
- Só suporta 64 bits.
- Os jogos produzidos são pesados, ou seja, exigem um bom Hardware.

### 2.4.3 GameMaker

GameMaker é um game engine, criado pela YoYo, mais indicado para iniciantes, ou seja, não é necessário saber programar nem entender muito da área de desenvolvimento de jogos para iniciar um projeto. Este vem fontes, texturas, sons, músicas entre outros. Para além disso permite exportar os jogos para diversas lojas, como a a Steam, Google Play, iOS App Store, Windows Store e Mac Store (Fidelis, 2018)..

Algumas vantagens são:

- A interface é simples, como já foi referido anteriormente.
- Permite desenvolver jogos sem saber programar.
- Permite scripts baseados na linguagem de programação.
- Suporta animação, possui um editor de imagens embutido.
- Permite exportar para várias plataformas.

No caso de desvantagens são as funções completas deste motor do jogo só estão disponíveis na versão paga, sendo estas muito caras.

### 2.4.4 RPG Maker

RPG Maker foi lançado no japão em 1998, em que a primeira versão chama-se RPG Maker 95, ao longo do tempo foram lançadas novas versões, sendo a mais recente a RPG Maker MV (Fidelis, 2018)..

Algumas vantagens da utilização do RPG Maker são, a utilização de scripts/eventos sem a necessidade de programar, possuir muitos tutoriais e uma comunidade muito ativa, possui vários gráficos disponíveis para importação, permite importar scripts, tem suporte Javascript (versão MV). No entanto o game *engine* em todas as versões é pago, não se consegue criar jogos para além dos modelos de RPG e só suporta 2D.

Algumas vantagens da utilização do RPG Maker são, a utilização de scripts/eventos sem a necessidade de programar, possuir muitos tutoriais e uma comunidade muito ativa, possui vários gráficos disponíveis para importação, permite importar scripts, tem suporte Javascript (versão MV). No entanto o game *engine* em todas as versões é pago, não se consegue criar jogos para além dos modelos de RPG e só suporta 2D.

## 2.5 Gestão de Projeto

Para gestão e planeamento do projeto serão utilizadas as seguintes ferramentas:

### 1. Github

Para perceber qual a função do Github, primeiro, é necessário entender o que se trata o Git (BROWN, 2019). Este é um sistema de controle de versões de código aberto, iniciado por Linus Torvalds, o mesmo criador do Linux.

O Git é uma ferramenta de linha de comandos, o github é um serviço online.

### 2. Trello

Trello é uma plataforma, que permite a organização e gestão das várias tarefas, que serão realizadas ao longo dos projetos pretendidos (Trello, s.d.).

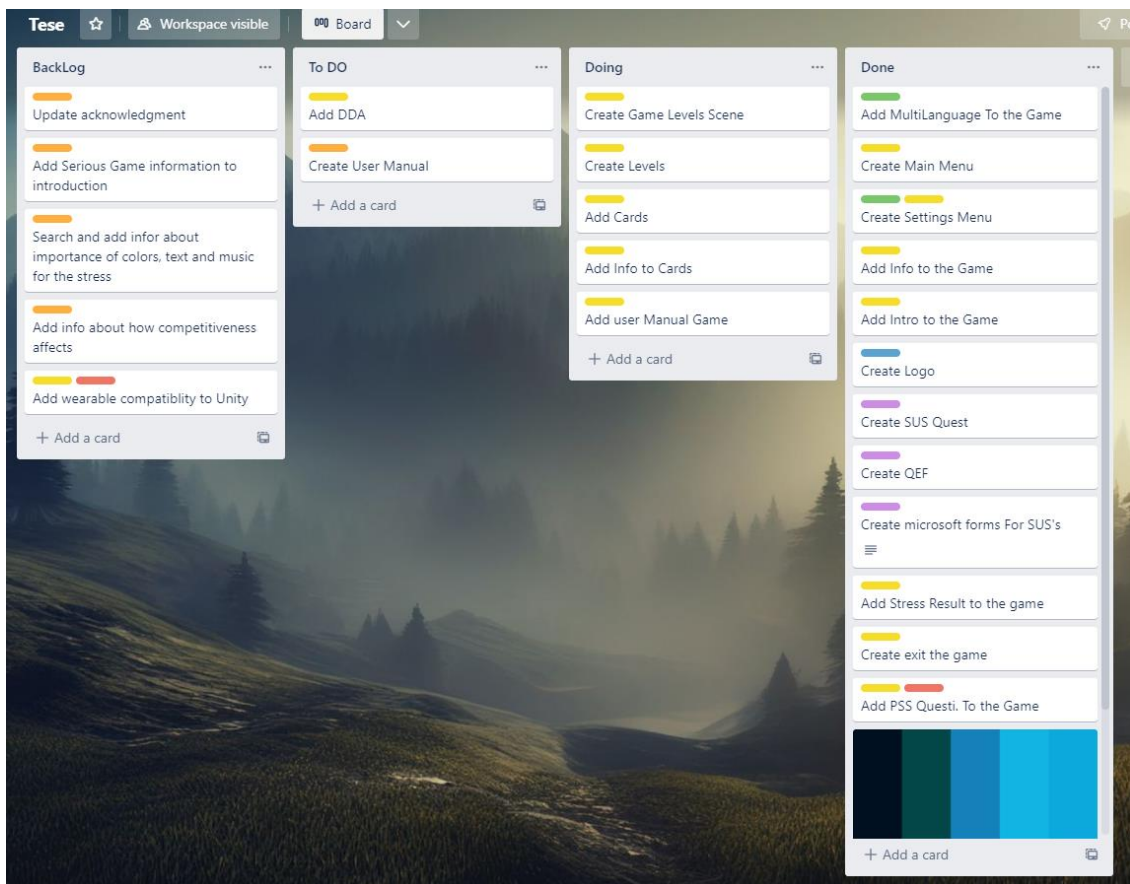


Figura 13. Trello projeto



## 3 Análise de valor

Neste capítulo é descrito o processo de análise de valor realizada para a oportunidade de negócio, em que este projeto se insere.

Primeiro foi realizada uma identificação e análise das oportunidades existentes. De seguida, de acordo com essas análises, foi realizada uma seleção de ideias e escolha da que melhor se adapta ao projeto, utilizando o método AHP. Tal como será apresentado o valor da solução, através de um CANVAS de proposta de valor, e no final realizou-se um QDF.

### 3.1 Modelo New Concept development

De acordo com Koen, o processo de inovação irá passar por três fases, nas quais The front end of innovation (FEI), New Product Development (NPD) e a fase de comercialização.

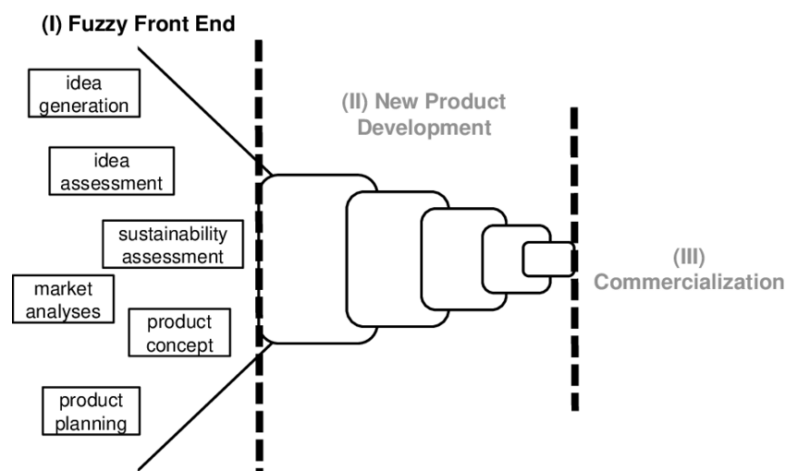


Figura 14. Processo de Inovação

A fase de **front-end of innovation** (FFE/FEI), a primeira fase, são atividades executadas antes do processo NPD (New Product Development). Esta fase representa um estado não estruturado, algo caótico e experimental, onde existem várias incertezas, de acordo com Koen (A.Koen, Ajamian, & Scott) trata-se de um momento “Eureka”.

A segunda fase, **New Product Development** (NPD), processo mais estruturado com um conjunto formalizado de atividades e perguntas.

De modo a otimizar a primeira fase, foi desenvolvido um novo modelo, chamado **New Concept development** (NCD). Este serve para fornecer uma linguagem comum entre a primeira e a segunda fase. É constituído por três partes, nas quais o motor, que representa a liderança, cultura e estratégia de uma organização, pela área interna, ou seja, os cinco elementos, nas quais, Identificação de Oportunidades, Análise de oportunidades, Geração e enriquecimento de ideias, seleção de ideias e definição do conceito, como se pode visualizar na Figura 15. Por último os fatores externos, que influenciam o motor, bem como altera o processo de inovação dos cinco elementos.

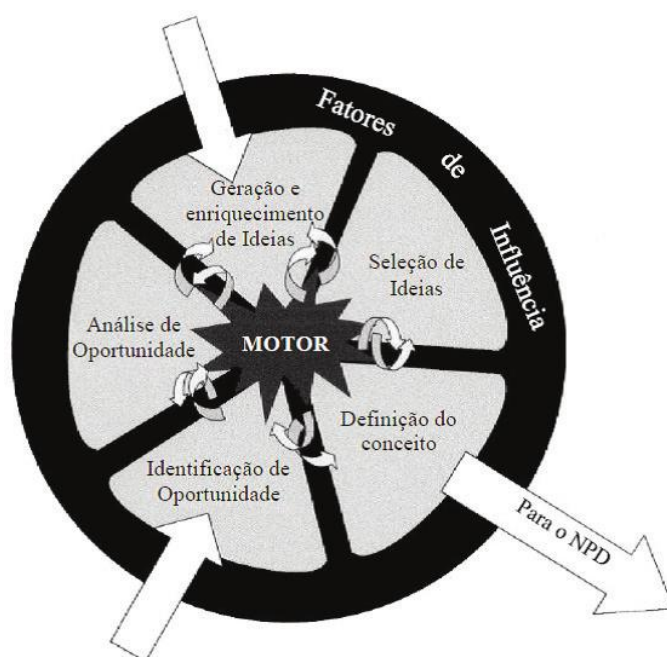


Figura 15. O Modelo New Concept Development

De seguida, foi feita uma comparação e análise entre o projeto a ser desenvolvido com os cinco elementos mencionados anteriormente.

#### 1. Identificação de Oportunidades

Esta fase, serve para identificar as derivadas oportunidades existentes, sendo que estas podem ser enquadradas em diversos contextos (A.Koen, Ajamian, & Scott). No caso prático, a identificação da oportunidade foi realizada pelo grupo de investigação GILT.

Sendo o Stress em ambiente empresarial um problema grave que afeta a produtividade do colaborador (Santos, s.d.), existe a necessidade de uma solução de modo a prevenir que este não aconteça, ou que o seu efeito não seja prejudicial, tal como, o fornecimento de informação aos colaboradores do que se trata o stress. E pela falta de um mecanismo que faço o descrito anteriormente, foi identificado como oportunidade o desenvolvimento de um jogo sério, de modo a prevenir o efeito do stress em ambiente empresarial.

O stress é uma das principais razões dos “*burnouts*”, termo utilizado para um esgotamento físico e mental, e com estes, os colaboradores de uma empresa poderão usufruir de baixa médica, isto levará a que a organização tenha menos colaboradores, tal como os rendimentos poderão baixar. Sendo assim, foi identificada outra oportunidade relacionada com a informação descrita anteriormente, ou seja, a implementação do jogo sério a ser desenvolvido, num ambiente empresarial, de modo a auxiliar o bem-estar dos colaboradores, para que estes se sintam bem, desta forma aumentando a produtividade dos mesmos e por consequência aumentar os rendimentos da empresa.

## 2. Análise de Oportunidades

Nesta fase, a oportunidade foi avaliada de modo a confirmar se seria benevolente persegui-la. Foi necessário por parte da autora do documento, analisar e obter informações adicionais, para passar a oportunidade para uma oportunidade específica em termos de integração no mercado e tecnologias. (A.Koen, Ajamian, & Scott)

De acordo com Koen, uma análise típica para uma oportunidade em grande escala seria necessária uma pesquisa sobre os seguintes pontos (A.Koen, Ajamian, & Scott):

- Enquadramento estratégico: identificar como esta oportunidade se encaixa dentro dos pontos fortes, fracos, ameaças e restantes oportunidades.
- Avaliação do segmento de mercado: Identificar uma descrição detalhada do segmento de mercado, demonstrando a razão pela qual este representa uma grande oportunidade.
- Análise de concorrentes: Determinar quem são os principais concorrentes no segmento de mercado identificado.
- Avaliação do cliente: Determina quais as principais necessidades do cliente que atualmente não são atendidos pelos produtos atuais.

Para a execução desta análise foi decidido utilizar uma análise SWOT. Esta é uma ferramenta (Casarotto, 2019), criada na década de 1960, por Albert S. Humphrey para planear a estratégia na gestão de projetos, é utilizada para analisar os cenários e auxiliar na toma de decisões.

Swot significa Strengths (Forças), Weaknesses (Fraquezas), Opportunities (Oportunidades) e Threats (Ameaças), é uma matriz 2x2 dividida nesses 4 pontos.

As forças e Fraquezas, são relacionadas com o ambiente interno, nas quais, é possível ter controlo, alguns exemplos são, o tempo de mercado, a localização de uma empresa, os recursos financeiros, entre outros.

As oportunidades e ameaças são relacionadas com o ambiente externo, ou seja, é onde entram questões relacionadas à política, concorrência, mudanças na legislação, entre muitas outras. É essencial nesta análise, estar atentos a tudo que envolve o mercado no qual o projeto está inserido.

Foi criada a Tabela 2, que representa uma matriz SWOT para análise do jogo sério a ser desenvolvido.

Tabela 2. Matriz SWOT

<p>Forças:</p> <ul style="list-style-type: none"> <li>• Ênfase a um assunto necessário ter em consideração;</li> <li>• Utilização de um jogo para abordar o assunto;</li> <li>• Produto final gratuito;</li> </ul>	<p>Fraquezas:</p> <ul style="list-style-type: none"> <li>• Investimento financeiro;</li> <li>• Equipa de desenvolvimento consiste em um elemento;</li> <li>• Falta de conhecimento sobre métodos eficazes para a prevenção do stress;</li> </ul>
<p>Oportunidades:</p> <ul style="list-style-type: none"> <li>• Reconhecimento por parte de outras organizações;</li> <li>• Implementação num ambiente empresarial;</li> </ul>	<p>Ameaças:</p> <ul style="list-style-type: none"> <li>• Mercado muito competitivo;</li> </ul>

### 3. Geração e enriquecimento de ideias

Esta, refere-se ao nascimento, desenvolvimento e amadurecimento de uma ideia concreta. A geração de ideias é evolutiva, estas são construídas, destruídas, combinadas, remodeladas, modificadas e atualizadas. Uma ideia pode passar por muitas iterações e mudanças à medida que é examinado o estudado, discutindo e desenvolvendo em conjunto com outros elementos. O contacto direto com clientes e utilizadores, bem como colaboração com outras empresas e instituições muitas vezes facilita esta atividade (A.Koen, Ajamian, & Scott).

Como este projeto, não possui parceiros de negócio, nem restantes colaboradores, a ideia central será desenvolver um jogo sério do tipo simulador/rpg, com puzzles.

### 4. Seleção de ideias

Nesta fase, como o nome indica, é a fase dedicada para a seleção da ideia que se pretende prosseguir. Como este projeto, não possui parceiros de negócio, como já referido anteriormente, faltava executar a escolha de motor de jogo que mais se adequa para o desenvolvimento deste jogo. Para tal foi decidido utilizar o método Analytic Hierarchy Process (AHP), que será apresentado no próximo subcapítulo.

### 3.1.1 Analytic Hierarchy Process

O Analytic hierarchy Process (AHP), é um método para organizar, analisar decisões complexas, utilizando cálculos matemáticos de modo a alcançar a melhor opção, desenvolvido por Thomas L. Este é utilizado como um meio de apoio a decisão, de modo a alcançar um determinado objetivo (Technology, s.d.).

#### 3.1.1.1 Construção da árvore hierárquica

O primeiro passo, foi construir a árvore hierárquica, para isso, necessário alinhar os seguintes pontos:

- Qual o objetivo ou finalidade da escolha. Neste caso será o motor de jogo a ser utilizado para o desenvolvimento do projeto. As alternativas escolhidas, já referidas no subcapítulo 0, são o unity, unreal engine, game maker e rpg maker.
- Quais os critérios essenciais para a escolha ser realizada. Estes foram:
  - **Preço:** Valor no caso de ser necessário comprar objetos ou recursos.
  - **Saturação:** Existência de muitos jogos desenvolvidos no mesmo motor de jogo.
  - **Gráficos:** A qualidade dos gráficos ou interfaces do motor de jogo.
  - **Motivação:** A predisposição para desenvolver o jogo num determinado motor de jogo.
  - **Material de apoio:** Consiste na quantidade de material de apoio disponível online.

Levando os pontos acima em consideração foi possível montar a árvore hierárquica representada na Figura 16.

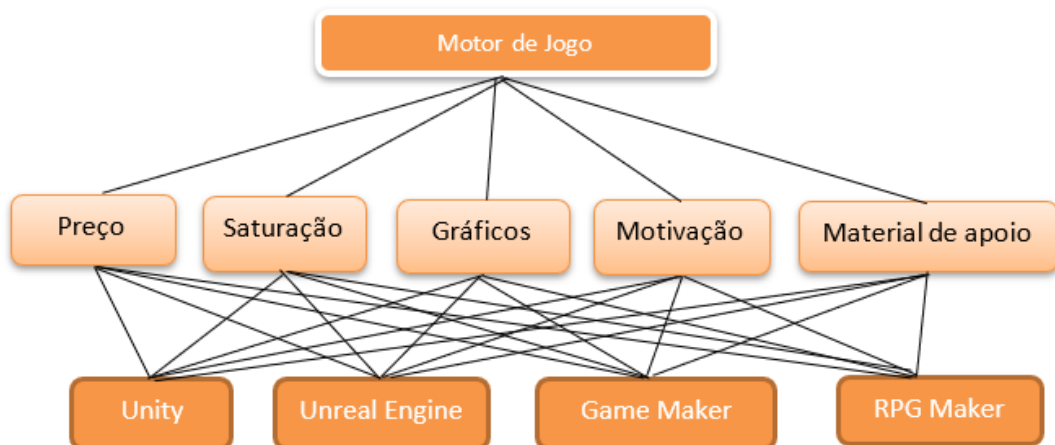


Figura 16. Árvore hierárquica

### 3.1.1.2 Comparação dos critérios

Nesta etapa, foram estabelecidas as prioridades entre os critérios escolhidos através de uma matriz de comparação. Para isso será utilizada a escala fundamental, representada na Figura 17.

Intensity of importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective
3	Moderate importance	Experience and judgement slightly favour one activity over another
5	Essential importance	Experience and judgement strongly favour one activity over another
7	Very strong importance	An activity is favoured very strongly over another; its dominance demonstrated in practice
9	Extreme importance	The evidence favouring one activity over another is of the highest possible order of affirmation
2, 4, 6, 8	Intermediate values	When compromise is needed between two

Figura 17. Escala Fundamental. (Joost, Esther, & Bahareh, 2015)

Tendo em consideração o seguinte valor, será necessário comparar os critérios entre eles mesmos de modo a obter uma matriz de comparação.

Tabela 3. Matriz de comparação dos critérios

Critérios	Preço	Saturação	Gráficos	Motivação	Material de Apoio
<b>Preço</b>	1	5	2	3	$\frac{1}{5}$
<b>Saturação</b>	$\frac{1}{5}$	1	2	$\frac{1}{5}$	$\frac{1}{4}$
<b>Gráficos</b>	$\frac{1}{2}$	$\frac{1}{2}$	1	$\frac{1}{4}$	$\frac{1}{2}$
<b>Motivação</b>	$\frac{1}{3}$	5	4	1	$\frac{1}{9}$
<b>Material de Apoio</b>	5	4	2	9	1

Após realizada a matriz de comparação para os critérios passou-se para o próximo passo descrito no subcapítulo seguinte.

### 3.1.1.3 Cálculo de Prioridade de cada critério

Tendo a matriz de comparação, passou-se para a próxima fase, nas quais seria necessário executar o somatório por coluna de cada critério obtendo os valores representados na Tabela 4.

Tabela 4. Somatório das colunas anteriores

<b>Critérios</b>	<b>Preço</b>	<b>Saturação</b>	<b>Gráficos</b>	<b>Motivação</b>	<b>Material de apoio</b>
<b>Total</b>	7.03	15.5	11	13.45	2.06

Tendo obtido os resultados anteriores, foi criada uma matriz de modo a representar a matriz normalizada dos critérios, obteve-se a matriz representada na Tabela 5.

Tabela 5. Matriz normalizada dos critérios

<b>Critérios</b>	<b>Preço</b>	<b>Saturação</b>	<b>Gráficos</b>	<b>Motivação</b>	<b>Material de apoio</b>
<b>Preço</b>	0,14	0,32	0,18	0,22	0,1
<b>Saturação</b>	0,03	0,06	0,18	0,01	0,12
<b>Gráficos</b>	0,07	0,03	0,09	0,02	0,24
<b>Motivação</b>	0,05	0,32	0,36	0,07	0,05
<b>Material de apoio</b>	0,71	0,26	0,18	0,67	0,49

Após obter a matriz normalizada dos critérios foi calculado a média aritmética dos valores para cada linha. O resultado irá indicar o vetor de prioridade.

Tabela 6. Peso dos critérios

<b>Critérios</b>	<b>Prioridade Relativa</b>
<b>Preço</b>	0,19
<b>Saturação</b>	0,08
<b>Gráficos</b>	0,09
<b>Motivação</b>	0,17
<b>Material de apoio</b>	0,46

Através da leitura da Tabela 6, é possível determinar a ordem de prioridade relativa de cada um dos critérios nas quais, Material de apoio > Preço > Motivação > Gráficos > Saturação.

Em suma, o critério mais importante, considerado para a autora, é o material de apoio, pois sem estes e sem conhecimentos aprofundados na utilização do motor de jogo, irá dificultar o desenvolvimento, podendo atrasar este. Caso haja mais material de apoio, a implementação e o desenvolvimento do jogo será mais fácil, de seguida o preço, para o desenvolvimento do projeto não se pretende usufruir de um orçamento, a motivação sendo os menos importantes a saturação e os gráficos.

#### 3.1.1.4 Avaliar a consistência das prioridades relativas

Nesta fase, foi avaliada consistência das prioridades, para isso foi necessário calcular a Razão de Consistência (RC), em que o RC é igual ao Índice de Consistência (IC) a dividir pelo Índice Aleatório (IR), segue-se a fórmula em (1).

$$RC = \frac{IC}{IR} \quad (2008, p. 4) \quad (1)$$

O IC em (2) é calculado pela seguinte formula:

$$IC = \frac{\lambda_{max} - n}{n-1} \quad (2008, p. 4) \quad (2)$$

Onde o  $\lambda_{max}$  representa o valor próprio da matriz e o  $n$  representa o número de critérios.

Em 1994, *Saaty* (Cleber & Mischel, 2008), propôs a seguinte tabela (Tabela 7) com os índices aleatórios em inglês *Random Index* (RI), para matrizes de ordem 1 a 10.

Tabela 7. Índices aleatórios

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.52	0.89	1.11	1.25	1.35	1.40	1.45	1.49

Neste caso o  $\lambda_{max}$  irá ser igual a 4.8, e o  $n$  igual a 5, sendo assim o cálculo do IC será igual a 0.05.

$$IC = \frac{4.8 - 5}{5-1} = 0.05 \quad RC = \frac{0.05}{1.11} = 0.04$$

Como o RC deu menor que zero, pode-se afirmar que os valores atribuídos as prioridades relativas são consistentes.

#### 3.1.1.5 Construção da matriz de comparação para cada critério

Durante esta etapa foram criadas as matrizes de comparação para cada um dos critérios definidos tendo enquanto as alternativas. Depois executado o somatório de cada coluna para utilizar esses ao gerar a matriz normalizada (valor de cada linha a dividir pela soma da coluna). Será demonstrado este processo para o primeiro critério, o preço nos restantes critérios será apresentado a matriz de comparação e o resultado do vetor.

Na Tabela 8 foi criada a matriz de comparação para o critério preço, após executado o somatório obteve-se o resultado descrito na Tabela 9.

Tabela 8. Matriz de comparação em relação ao Preço

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	1	5	5	5
Unreal Engine	$\frac{1}{5}$	1	2	3
GameMaker	$\frac{1}{5}$	$\frac{1}{2}$	1	2
RPG Maker	$\frac{1}{5}$	$\frac{1}{3}$	$\frac{1}{2}$	1

Tabela 9. Somatório das colunas anteriores de preço

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Total	1.6	6.83	8.5	11

Foram feitos os dados cálculos necessários para obter a matriz normalizada, como é possível verificar na Tabela 10.

Tabela 10. Matriz normalizada das opções em relação com o Preço

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	0,62	0,73	0,59	0,45
Unreal Engine	0,13	0,15	0,24	0,27
GameMaker	0,13	0,07	0,12	0,18
RPG Maker	0,13	0,05	0,06	0,09

Foram obtidos os seguintes cálculos finais, que criam o vetor prioridade em relação ao preço Tabela 11.

Tabela 11. Vetor prioridade em relação ao Preço

Opções	Vetor prioridade
Unity	0.6
Unreal Engine	0.2
GameMaker	0.12
RPG Maker	0.1

Na Tabela 12, foi criada a matriz de comparação para o critério saturação.

Tabela 12. Matriz de comparação em relação à Saturação

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	1	2	4	$\frac{1}{5}$
Unreal Engine	$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{4}$
GameMaker	$\frac{1}{4}$	2	1	$\frac{1}{4}$
RPG Maker	5	4	4	1

Foram obtidos os seguintes cálculos finais, que criam o vetor prioridade em relação à saturação Tabela 13.

Tabela 13. Vetor prioridade em relação à Saturação

Opções	Vetor prioridade
Unity	0,23
Unreal Engine	0,1
GameMaker	0,13
RPG Maker	0,55

Na Tabela 14 foi criada a matriz de comparação para o critério gráficos.

Tabela 14. Matriz de comparação em relação aos Gráficos

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	1	5	5	5
Unreal Engine	$\frac{1}{5}$	1	2	3
GameMaker	$\frac{1}{5}$	$\frac{1}{2}$	1	2
RPG Maker	$\frac{1}{5}$	$\frac{1}{3}$	$\frac{1}{2}$	1

Foram obtidos os seguintes cálculos finais, que criam o vetor prioridade em relação aos gráficos Tabela 15.

Tabela 15. Vetor prioridade em relação ao Gráficos

Opções	Vetor prioridade
Unity	0,6
Unreal Engine	0,2
GameMaker	0,12
RPG Maker	0,1

Na Tabela 16 foi criada a matriz de comparação para o critério motivação.

Tabela 16. Matriz de comparação em relação à Motivação

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	1	5	5	5
Unreal Engine	$\frac{1}{5}$	1	3	3
GameMaker	$\frac{1}{5}$	$\frac{1}{3}$	1	$\frac{1}{5}$
RPG Maker	$\frac{1}{5}$	$\frac{1}{3}$	5	1

Foram obtidos os seguintes cálculos finais, que criam o vetor prioridade em relação à motivação, como pode ser verificado na Tabela 17.

Tabela 17. Vetor prioridade em relação à Motivação

Opções	Vetor prioridade
Unity	0,57
Unreal Engine	0,2
GameMaker	0,07
RPG Maker	0,16

Na Tabela 18 foi criada a matriz de comparação para o critério material de apoio.

Tabela 18. Matriz de comparação em relação ao Material de Apoio

Opções	Unity	Unreal Engine	GameMaker	RPG Maker
Unity	1	5	5	3
Unreal Engine	$\frac{1}{5}$	1	4	$\frac{1}{3}$
GameMaker	$\frac{1}{5}$	$\frac{1}{4}$	1	$\frac{1}{3}$
RPG Maker	$\frac{1}{3}$	3	3	1

Foram obtidos os seguintes cálculos finais, que criam o vetor prioridade em relação ao material de apoio, como pode ser verificado na

Tabela 19.

Tabela 19. Vetor prioridade em relação ao Material de Apoio

Opções	Vetor prioridade
Unity	0,54
Unreal Engine	0,15
GameMaker	0,08
RPG Maker	0,24

Para terminar este passo, foi decidido construir um diagrama, como se pode verificar na Figura 18. Com toda a informação obtida nos cálculos anteriores, ou seja, o resultado dos vetores de prioridade.

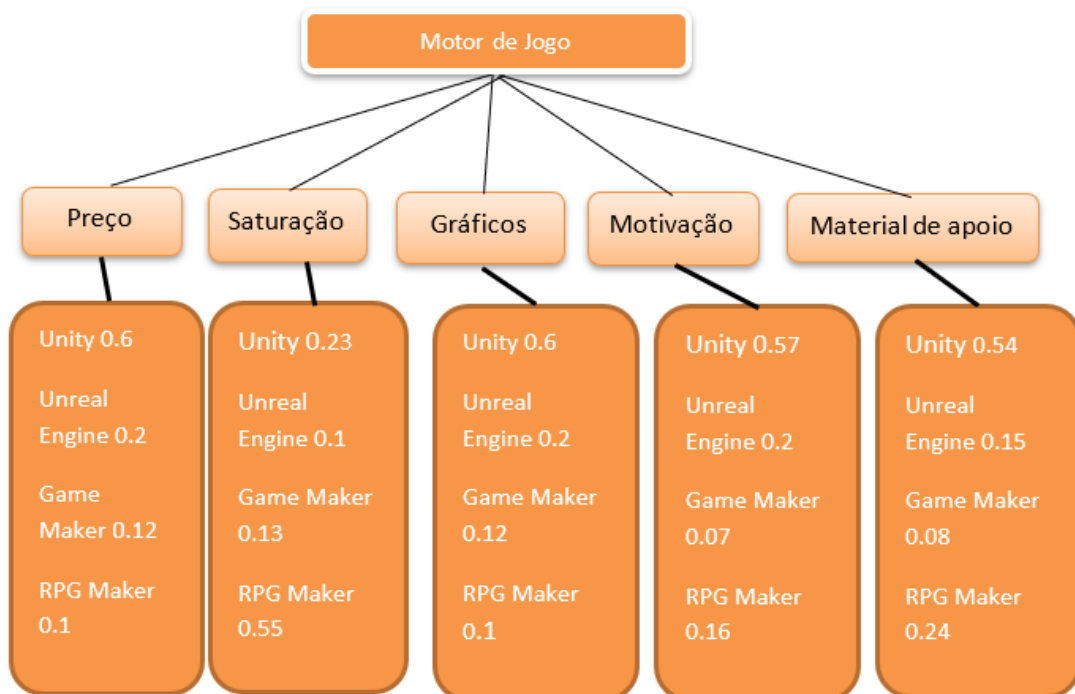


Figura 18. Diagrama com os resultados

### 3.1.1.6 Resultado

Esta é o último passo do método AHP, que irá utilizar os valores anteriores para demonstrar qual a melhor opção. Juntando os valores anteriores obtêm-se o seguinte cálculo:

$$\begin{bmatrix} 0.6 & 0.23 & 0.6 & 0.57 & 0.54 \\ 0.2 & 0.1 & 0.2 & 0.2 & 0.15 \\ 0.12 & 0.13 & 0.12 & 0.07 & 0.08 \\ 0.1 & 0.55 & 0.1 & 0.16 & 0.24 \end{bmatrix} \times \begin{bmatrix} 0.19 \\ 0.08 \\ 0.09 \\ 0.17 \\ 0.46 \end{bmatrix} = \begin{bmatrix} 0.53 \\ 0.17 \\ 0.09 \\ 0.21 \end{bmatrix}$$

Em suma, a melhor opção será a primeira com o valor de 0.53, sendo assim o motor de jogo que irá ser utilizado para o desenvolvimento do projeto é o Unity.

## 3.2 Valor da solução

Depois das análises realizadas nos capítulos anteriores, pretende-se, já se ter respondido as seguintes questões, ao longo do documento, nas quais:

- “What do you do” ou O que será feito?
- “Who cares” ou Qual a importância?
- “Why you choose the idea” ou Justificação da escolha?
- “Target Customer” ou Quem são os clientes ou utilizadores?
- “What value for them?” ou Qual o valor para estes Utilizadores?

Em suma, o objetivo principal é desenvolver um jogo com o objetivo de sensibilizar os colaboradores de uma organização com a importância de evitar o stress, tal como auxiliar na prevenção deste. O Stress é algo negativo que poderá provocar doenças graves caso não seja controlado, sendo assim será importante a ênfase neste tema. Foi executada a escolha deste projeto, devido à autora deste documento, ser colaboradora numa organização, e saber, por experiência própria, quais as dificuldades que passam os restantes. A ideia seria para implementar numa organização, em que todos os colaboradores pudessem aceder, através de um aparelho móvel. Esta ideia iria permitir aos utilizadores terem mais conhecimentos sobre o tema, ser um mecanismo de distração e melhorar possivelmente a sua saúde.

### 3.2.1 Proposta de valor

Uma proposta de valor pode ser entendida como a maneira que uma empresa consegue representar os benefícios empregues nos seus produtos ou serviços, para os clientes que os irão adquirir. Esta é uma estratégia de marketing que pretende indicar a relação entre empresa e a decisão de compra dos clientes (Proposta de valor, s.d.).

Foi aplicada uma proposta de valor para a solução, através de um Canvas. Este é dividido em 2 partes, nas quais no perfil do cliente ou utilizador e no mapa de valor (Canvas de Proposta de Valor: passo a passo para criar o seu, 2020). No perfil do cliente contem os seguintes campos, para preencher com informação (Pereira D. , 2019):

- Ganhos - Resultados que o cliente pode vir a obter pela utilização do produto.
- Tarefas – Tarefas que o cliente está a tentar executar ou até mesmo problemas que estão a tentar resolver.
- Dores – Descrever as emoções que o cliente pode ter ou esteja insatisfeito ao longo a realização das tarefas anteriores.

Na parte, mapa de valor, ou solução, este possui:

- Produtos e serviços - Pretende-se listar todos os produtos e serviços que a proposta de valor da solução está baseada.
- Aliviador de Dores – Explicar como a solução irá agradar às dores relacionadas com o cliente.
- Criadores de Ganhos – Descrever como a solução fornecerá ganhos aos clientes.

Relacionado com a solução a desenvolver e com o que foi descrito anteriormente elaborou-se o seguinte Canvas para a proposta de valor.

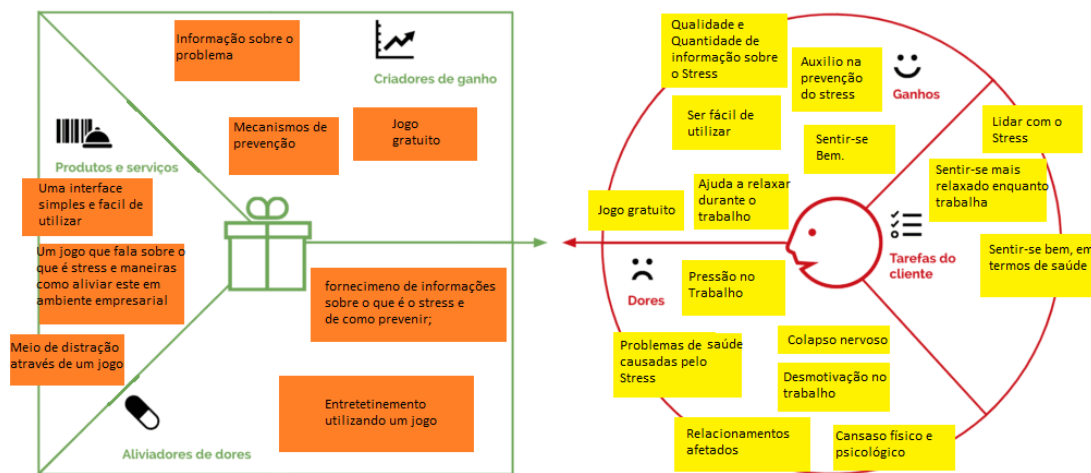


Figura 19. Canvas da proposta de valor

### 3.2.2 Quality Function Deployment

O *Quality Function Deployment*, ou QFD, é uma ferramenta, criada pelo japonês Yoji Akao, que permite que uma equipa de desenvolvimento de um produto, utilize as necessidades do cliente e incorporem nos projetos de modo a implementar uma melhoria. Este corresponde a 4 matrizes, onde é executado o planeamento do produto. Esta fase chama-se Casa da Qualidade ou *House of Quality* ou HOQ (Desdobramento da Função Qualidade (QFD), s.d.).

Na HOQ, pretende-se identificar o que o cliente pretende, identificar como o produto irá satisfazer as necessidades do cliente, identificar as relações entre os como irá ser realizado, desenvolver uma classificação de importância, avaliar produtos ou serviços concorrentes e por fim determinar os atributos técnicos desejáveis.

Foi aplicado o descrito acima, e realizado um HOQ em contexto do projeto a ser realizado. Primeira fase, consistiu na definição do que o cliente ou utilizador necessita, neste caso, alcançaram-se as seguintes necessidades:

- Jogo simples;
- Mecanismo para relaxar e distrair;
- Mecanismo para prevenir o stress;
- Informações sobre o stress;
- Interfaces chamativas;

Tendo as necessidades do cliente na coluna respetiva (verificar Figura 21, coluna *Demanded Quality*) foi dada a continuação do processo, nas quais atribuídos valores de importância, estes derivam de 1 a 5, onde 1 representa mínima importância e 5 máxima importância. Depois dessa atribuição foi calculada a sua normalização, executada automaticamente após preenchido template em Excel, disponibilizado online (House Of Quality).

O próximo passo, foi identificar as características de qualidade ou maneiras de como iriam responder às necessidades dos clientes (verificar Figura 21, linha *Quality Characteristics*), nas quais se segue a lista:

- Interface gráfica simples;
- Jogo com um DDA adequado;
- Aparecimento de informações sobre o stress;
- Aparecimento de possíveis métodos para ajudar a prevenir o stress;
- Feedback, se o jogo está a transmitir informações ao utilizador sobre os níveis de stress.

Após o descrito anteriormente, foi avaliado qual o grau de relações entre os *Demanded Quality* e os *Quality Characteristics*, ou seja, entre as necessidades do cliente, com as características de qualidade, utilizando símbolos respetivos aos valores 0 (Sem relação), 1 (Relação Fraca), 3 (Relação Média) e 9 (Relação Forte), como se pode verificar na Figura 20. Foi também calculado o peso relativo de cada característica de qualidade e preenchido a secção das correlações (Telhado da Casa), entre as características de qualidade, utilizando os símbolos respetivos a esta, pode-se verificar estes, novamente, na Figura 20.

Legenda		
⊕	Relação Forte	9
○	Relação Média	3
▲	Relação Fraca	1
	Sem relação	0
++	Correlação Fortemente Positiva	
+	Correlação Positiva	
-	Correlação Negativa	
▼	Correlação Fortemente Negativa	
	Sem Correlação	
▼	Objetivo é Minimizar	
▲	Objetivo é Maximizar	
X	Objetivo é acertar o alvo	

Figura 20. Legenda dos Valores e símbolos

Por último, foram comparados alguns jogos e a solução idealizada com o que o cliente pretende, foram usados valores de 0 a 5, onde 0 implica muito fraco, e 5 o melhor.

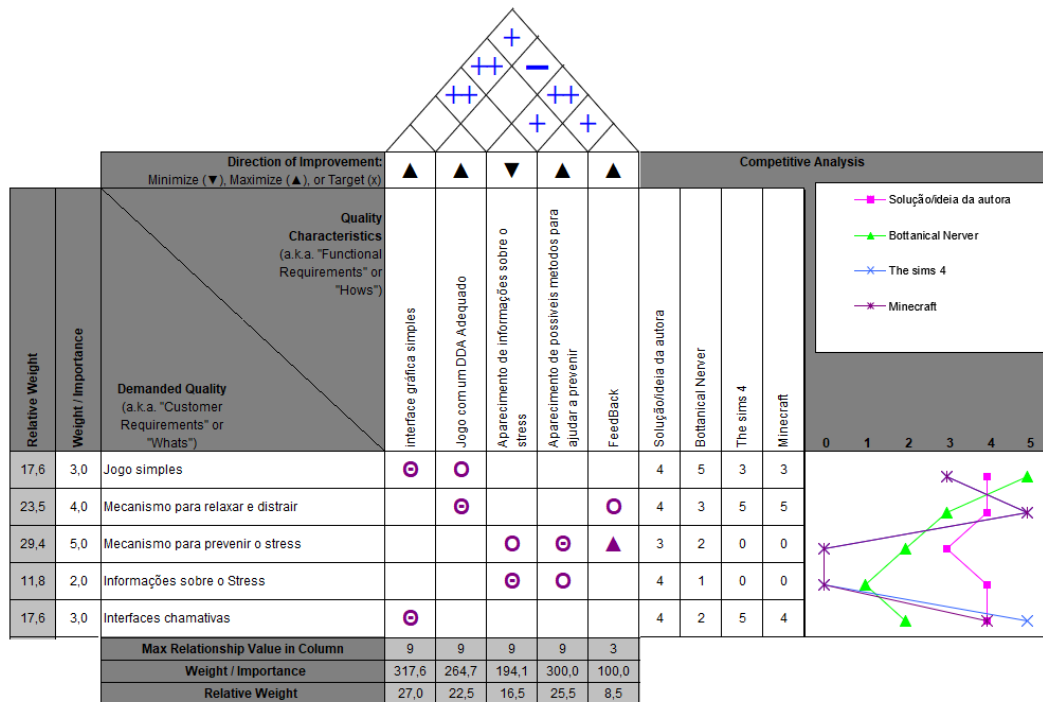


Figura 21. House of Quality

Através da análise da Figura 21, é possível obter as seguintes informações:

- O requisito **Informações sobre o stress** é o que possui menor grau de importância;
- O requisito **Mecanismo para prevenir o stress** é o que possui maior grau de importância com 29%, sendo o Mecanismo para relaxar e distrair o segundo, com cerca de 24%;
- O requisito **Jogo simples** e **Interfaces aclamativas** tem o mesmo grau de importância;
- A característica de qualidade **Interface gráfica simples** possui uma correlação fortemente positiva com **Aparecimento de informações sobre o stress**, com aparecimento de possíveis métodos para ajudar a prevenir;
- A característica de qualidade Interface gráfica simples possui uma correlação positiva com o Feedback;



## 4 Avaliação de Abordagens

Neste capítulo será executada uma análise e avaliação aos métodos que auxiliam na prevenção e alívio do stress, tal como uma avaliação sobre a tecnologia escolhida, se esta se adapta, de modo a cumprir o objetivo do jogo sério a ser desenvolvido.

### 4.1 Stress e Abordagens

Como referido no capítulo 2.1 o stress é um problema que afeta muitos colaboradores, muitas das vezes por falta de organização do material de trabalho, excesso de tarefas, entre outros fatores. Portanto, de seguida serão descritos alguns métodos para um colaborador conseguir aliviar o stress, tal como conceitos essenciais para o evitar.

De acordo com o artigo no site e-konomistas (Silva, 2017), escrito por Inês Silva, pesquisadores chegaram a conclusão de que a chave para diminuir o stress no local de trabalho são os jogos de vídeo. Um estudo foi realizado pela *Human Factors and Ergonomics Society*, que examinou 66 trabalhadores divididos em 3 grupos após algum tempo de estarem a trabalhar. Os membros do primeiro grupo passaram cinco minutos numa sala silenciosa, ou seja, sem computador e telemóveis. No segundo foram praticar exercícios de relaxamento e o restante grupo jogaram um jogo simples como Sushi Cat.

Sendo o último grupo com melhores resultados a lidar com o stress do trabalho. Sendo assim muitas empresas como a Blip, Altran Portugal, Bold by Devoteam, entre muitas outras, acabaram por criar uma sala de jogos para os colaboradores conseguirem aliviar o stress e aumentar a criatividade, níveis de concentração e produtividade (Portugal, s.d.).

Como muitas empresas, não adaptaram o descrito anteriormente, seguem-se algumas maneiras de aliviar o stress, sendo estes possíveis conhecimentos a abranger no jogo sério a ser desenvolvido:

- Preparar antecipadamente: pensar antecipadamente, nem sempre é possível um colaborador estar preparado para enfrentar uma situação de grande pressão, sendo assim este poderá pensar em maneiras de lidar com esta de modo a preparar-se.
- Aprender a dizer que “Não” : muitas ocasiões o colaborador já tem uma quantidade excessiva de tarefas, sendo assim a importância do dizer que “não”.
- Manter a calma: O colaborador manter a calma quando tem um obstáculo ou dificuldade para ultrapassar. Manter a calma irá facilitar no processo de análise e resolução de problemas.
- Pensamento Positivo: É normal um colaborador cometer erros, mas deverá aprender com estes, de modo a melhorar o seu desempenho.
- Comunicação: Muitas ocasiões o colaborador possui dúvidas, problemas numa determinada tarefa, sendo assim a importância da comunicação, tanto na lateral (com colaboradores iguais na hierarquia) como na ascendente (comunicação com colaboradores superiores).
- Valorize o tempo de descanso: Muitos colaboradores acabam por fazer mais do que o horário de trabalho. Sendo assim acabam por ficar cansados psicologicamente, este pode ser pior que desgasto físico.

Para determinar se o jogo sério desenvolvido está a cumprir um dos objetivos principais, já referidos no capítulo 1.1, poderá ser executada uma comunicação do jogo com um wearable mencionado no subcapítulo 2.1.3.

## 4.2 Motor de jogo selecionado

Em relação à tecnologia escolhida no subcapítulo 3.1.1, o Unity, possui um conjunto de características (Eduardo, 2015) que o levam a ser o melhor motor de jogo, de modo a alcançar o objetivo pretendido com o jogo sério. Segue-se a lista dessas características.

- Produção rápida: existência de uma excelente *pipeline* de recursos, os testes e a edição dos jogos é de modo imediato, fácil acesso à comunidade, documentação e *Asset Store*.
- Som e gráficos cinemáticos: possui opções de luz de alta qualidade e alto desempenho, possui efeitos especiais pós-processamento e sistemas de partículas.
- Otimização do desempenho: o unity disponibiliza *frame rate* suave, possibilidade de otimizar a jogabilidade com o *Unity Profiles*.
- Documentação e Tutoriais: o site é composto por uma elevada gama de tutoriais, disponibiliza também projetos prontos de modo a fornecer material de estudo, tal como vídeos dos criadores destes.
- Comunidade: é o ponto mais forte, pois existe uma comunidade ativa não só no fórum do próprio site, mas também no Facebook, wiki's e comunidade em aberto.
- *Asset Store*: disponibiliza modelos 3D, códigos, interfaces, texturas e até projetos já desenvolvidos.



Figura 22. Características Unity (Eduardo, 2015)



# 5 Desenho

Neste capítulo será apresentado o desenho conceptual do projeto a ser desenvolvido, tendo em consideração os padrões e regras adaptando boas práticas e ciências de engenharia informática.

## 5.1 Género de jogo escolhido

Foi decidido desenvolver um jogo do género de puzzle, um jogo de exemplo será o *candy crush*. Planeia-se desenvolver um jogo sério parecido com o exemplo indicado, visto que, pela análise feita no subcapítulo 2.2.2.4 este género de jogo é menos stressante e ajuda a aliviar o stress.

O jogo não terá um fator competitivo, não será um jogo online, nem irá funcionar com sistema de pontuação visível para o jogador.

## 5.2 Dificuldade

Neste subcapítulo o objetivo é descrever como será implementado o DDA no jogo. Pretende-se que à medida que o jogador for aumentando de nível e o cálculo do stress seja baixo, o grau de dificuldade vá variando de acordo com o tempo que o jogador demora a resolver o puzzle anterior, este tempo será transformado em pontos, estes não serão visíveis, será apenas para efeitos de cálculos. Para além disso será implementado maior número de peças ao longo dos níveis.

Supondo o seguinte cenário, dois jogadores, um que pela resposta do questionário obteve stress elevado, e o segundo, stress baixo. O primeiro nível será igual para ambos, os seguintes terão o mesmo número de peças para todos os jogares, mas com uma facilidade de resolução correspondente ao nível de stress do jogador. Para tal decisão considera-se que um jogador

mais stressado levará mais tempo a resolver um puzzle. Portanto, um jogador A, mais stressado, terá um segundo nível com uma mistura de peças inferior de modo facilitar a resolução do puzzle num menor espaço de tempo. Por comparação, um jogador B, com stress baixo, terá um segundo nível com uma maior mistura de peças e um método de baralhar diferente, dificultando assim a solução do puzzle.

Para isto se concretizar é necessário definir o seguinte:

- Quais os pontos que correspondem a intervalos de tempo, segue-se um exemplo representado na Tabela 20.

Tabela 20. Pontuação por tempo

Tempo/minutos	Valor dos pontos
Menos de 1	100 pontos
Entre 1 e 3	50 pontos
Entre 3 e 4	25 pontos
Mais de 4	0 pontos

- Quantidade de peças que cada nível irá ter, tendo em consideração que a cada 5 níveis esta quantidade iria reiniciar (Tabela 21).

Tabela 21. Relação número de peças e nível

Nível	Número de peças
1	4 (2x2)
2	9 (3x3)
3	16 (4x4)
4	24 (6x4)
5	25 (5x5)

- Sistema de baralho em que varia em baixo, médio ou elevado.

### 5.3 Levantamento de requisitos

Neste subcapítulo é executado um levantamento dos requisitos do projeto, tendo em consideração o problema analisado, sendo este já mencionado até esta fase do documento. De seguida estes foram separados em requisitos funcionais e não funcionais.

Seguem-se os requisitos levantados:

- R01: Disponível para múltiplas plataformas e dispositivos.
- R02: O jogo deve conter um menu principal.
- R03: O jogo deve conter um mapa de níveis.
- R04: O jogo deverá ter níveis para o jogador seleccionar qual pretende jogar.

- R05: O jogo deverá ter uma coleção de cartas, onde o jogador poderá ver mais informação destas.
- R06: O jogo deverá fornecer o questionário PSS (32 Perceived Stress Scale), tal como os resultados obtidos deste.
- R07: O jogo deverá ter opção de sair.
- R08: O jogo deverá possuir um manual de instruções.
- R09: O jogo deverá conter informações sobre este mesmo.
- R10: O jogo deverá ter consistência de cores.
- R11: Consistência no mecanismo de navegação.
- R12: O jogo deverá ser suportado em mais do que uma linguagem.
- R13: O jogo deverá permitir modificar o volume do áudio.
- R14: Todo o conteúdo deverá estar relacionado com o jogo.
- R15: O jogo deverá ter uma interface intuitiva.
- R16: O jogador deverá ter controlo pelas ações.
- R17: O jogo deverá promover a aprendizagem.
- R18: Os conceitos têm de ser apresentados de forma clara e oportuna.

### 5.3.1 Requisitos funcionais

Os requisitos funcionais (RF), de acordo com a norma IEEE-90 (Teixeira, 2018), são capacidades que o utilizador necessita para a resolução de um problema ou para atingir um objetivo. Sendo assim, foram selecionados os requisitos funcionais da lista referida anteriormente, e separados de acordo com os seguintes módulos.

- Definições: agrupa os requisitos que fazem parte alterações do jogo.
- Jogo: os requisitos relacionados com o jogo em si.
- Conhecimentos: agrupa os requisitos que pretendem fornecer informações ao jogador.

Nas seguintes tabelas são descritos os requisitos funcionais, fornecendo informação sobre o grau de prioridade, dependências com outros requisitos, e uma breve descrição.

Tabela 22. Requisito Funcional Número 1.

<b>Identificador</b>	R02: O jogo deve conter um menu principal.		
<b>Modulo</b>	Jogo	<b>Dependências</b>	Não aplicado
<b>Descrição</b>	O jogador deverá ter acesso a um menu principal, que irá direcionar para o resto das opções do jogo.		

Tabela 23. Requisito Funcional Número 2.

<b>Identificador</b>	R03: O jogo deve conter um mapa de níveis.		
<b>Modulo</b>	Jogo	<b>Dependências</b>	R02
<b>Descrição</b>	O jogador poderá aceder a um menu com os níveis disponíveis.		

Tabela 24. Requisito Funcional Número 3.

<b>Identificador</b>	R04: O jogo deverá ter níveis para o jogador selecionar qual pretende jogar.		
<b>Modulo</b>	Jogo	<b>Dependências</b>	R03
<b>Descrição</b>	O jogador poderá escolher um dos níveis, se já tiver desbloqueado para jogar.		

Tabela 25. Requisito Funcional Número 4.

<b>Identificador</b>	R05: O jogo deverá ter uma coleção de cartas, onde o jogador poderá ver mais informação destas.		
<b>Modulo</b>	Conhecimentos	<b>Dependências</b>	R04
<b>Descrição</b>	O jogador poderá ver informação das cartas que ganhou ao vencer num determinado nível.		

Tabela 26. Requisito Funcional Número 5.

<b>Identificador</b>	R06: O jogo deverá fornecer o questionário PSS (32 Perceived Stress Scale), tal como os resultados obtidos deste.		
<b>Modulo</b>	Conhecimentos	<b>Dependências</b>	Não aplicado
<b>Descrição</b>	O jogador poderá executar o questionário e verificar o resultado destes, de modo a obter o seu nível de stress.		

Tabela 27. Requisito Funcional Número 6.

<b>Identificador</b>	R08: O jogo deverá possuir um manual de instruções.		
<b>Modulo</b>	Conhecimentos	<b>Dependências</b>	Todos os requisitos
<b>Descrição</b>	O jogador poderá aceder a um manual de instruções.		

Tabela 28. Requisito Funcional Número 7.

<b>Identificador</b>	R09: O jogo deverá conter informações sobre este mesmo.		
<b>Modulo</b>	Conhecimentos	<b>Dependências</b>	Não aplicado
<b>Descrição</b>	O jogador poderá aceder a informações sobre o jogo.		

Tabela 29. Requisito Funcional Número 8.

<b>Identificador</b>	R12: O jogo deverá ser suportado em mais do que uma linguagem.		
<b>Modulo</b>	Definições	<b>Dependências</b>	Não aplicado
<b>Descrição</b>	O jogo deverá permitir ao utilizador para alterar a linguagem do jogo.		

Tabela 30. Requisito Funcional Número 9.

<b>Identificador</b>	R13: O jogo deverá permitir modificar o volume do áudio.		
<b>Modulo</b>	Definições	<b>Dependências</b>	Não aplicado
<b>Descrição</b>	O jogo deverá permitir ao utilizador alterar o volume do jogo.		

Tabela 31. Requisito Funcional Número 10.

<b>Identificador</b>	R07: O jogo deverá ter opção de sair.		
<b>Modulo</b>	Jogo	<b>Dependências</b>	
<b>Descrição</b>	Possibilidade de sair do jogo, sem perder os dados deste.		

Tabela 32. Requisito Funcional Número 11

<b>Identificador</b>	R19: O jogo deverá fornecer um botão de ajuda.		
<b>Modulo</b>	Jogo	<b>Dependências</b>	RF01 e RF02
<b>Descrição</b>	Botão de auxílio, de modo a ajudar o jogador a ver a jogada que deverá realizar.		

### 5.3.2 Requisitos não funcionais

Os requisitos não funcionais são um conjunto de elementos que descrevem as capacidades e, ou, restrições de um projeto. Estes são os requisitos que irão definir qual a qualidade de um projeto (altexsoft). Alguns exemplos são a velocidade, integridade de dados, harmonia de cores, entre muitos outros. Segue-se a lista dos requisitos não funcionais determinados na lista já referida:

- R01: Disponível para múltiplas plataformas e dispositivos.
- R10: O jogo deverá ter consistência de cores.
- R11: Consistência no mecanismo de navegação.
- R14: Todo o conteúdo deverá estar relacionado com o jogo.
- R15: O jogo deverá ter uma interface intuitiva.
- R16: O jogador deverá ter controlo pelas ações.
- R17: O jogo deverá promover a aprendizagem.
- R18: Os conceitos têm de ser apresentados de forma clara e oportuna.

## 5.4 Casos de Uso

Neste subcapítulo, foi elaborado um diagrama de casos de uso (Figura 23), tendo em consideração os dados do subcapítulo Requisitos funcionais. Este relaciona o ator, neste caso o jogador, com o sistema, ou seja, o jogo sério, nas quais fornece uma visão geral das ações que o jogador poderá realizar em relação ao jogo.

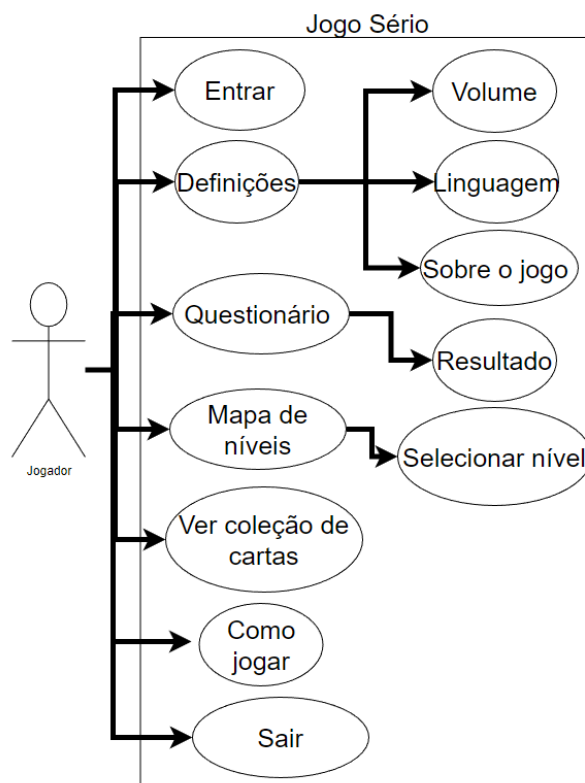


Figura 23. Diagrama de casos de Uso

## 5.5 Fluxograma

Foi criado um fluxograma (Figura 24) de modo a dar uma visão da solução a ser implementada. O esquema permite fornecer uma ideia das interações disponíveis para o jogador, tendo em conta os casos de uso já definidos (subcapítulo 5.3.2).

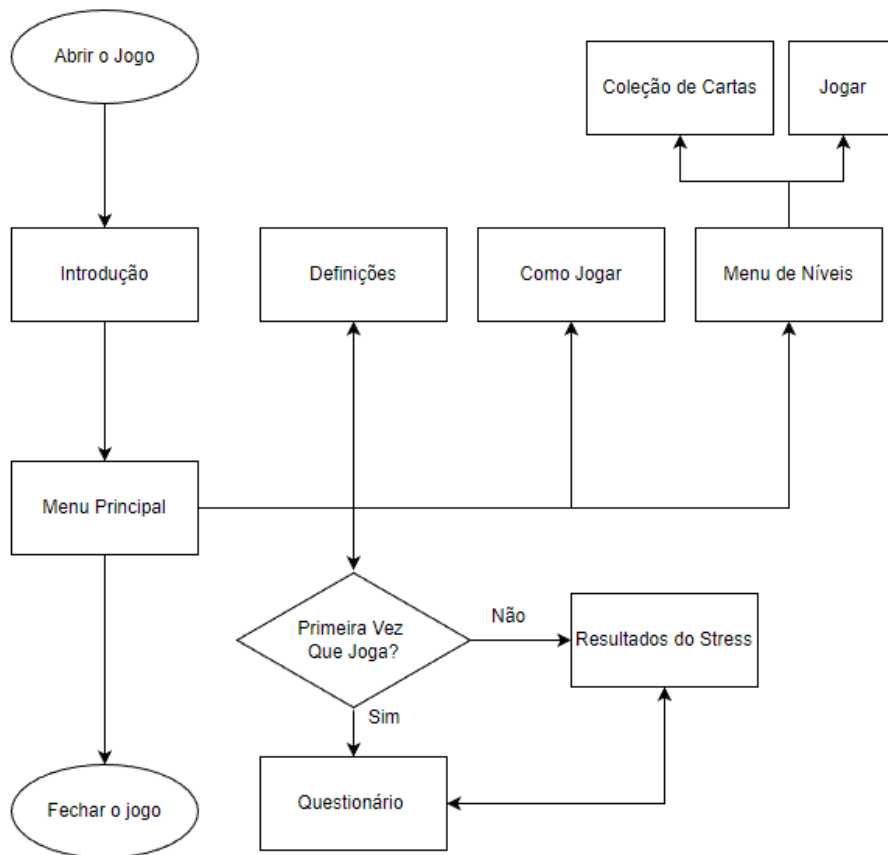


Figura 24. Fluxograma.

Quando um jogador entra na aplicação, irá aparecer uma introdução, com o logo do jogo, sendo depois reencaminhado para um menu principal, este será o mecanismo principal para a navegação do jogo, através da qual poderá aceder as definições, onde poderá alterar o idioma, configurar o volume do jogo e ver informações sobre este. Este poderá também ao seleccionar a opção “como jogar” ver informações sobre o funcionamento do jogo, também poderá ver os resultados do questionário PSS, e realizá-lo.

No caso de o jogador pretender jogar, será direccionado para um menu com níveis, onde ao seleccionar um nível poderá jogar. Os níveis serão desbloqueados à medida que o jogador vá ganhando. Neste mesmo menu dos níveis será possível ver as cartas que ganha ao longo destes.

Ainda no menu principal o jogador terá a opção de desligar o jogo, saindo deste.

## 5.6 Base de dados

Em relação à Base de dados (BD) a ser utilizada neste projeto, foi decidido optar pela *MongoDB Atlas*. Uma BD em *cloud*, desenvolvida pelas mesmas pessoas que criaram o *MongoDB* (MongoDB Atlas, s.d.). Tal como serão guardadas algumas informações preferenciais, recorrendo ao recurso *PlayerPrefs*. Este é uma classe auxiliar da Unity3D que serve para salvar as preferências de um jogador entre sessões do jogo, salvar configurações como linguagem do jogo, som habilitado, entre outros (Fernandes, 2016).

Foi criado um esquema em *json* (*JavaScript Object Notation*), de modo a demonstrar a estrutura de dados que será implementada no jogo.

```
{
  "$schema": "http://json-schema.org/draft-06/schema#",
  "description": "Database Schema of the game in develop",
  "type": "object",
  "properties": {
    "Cards": {
      "$ref": "#/definitions/Cards"
    },
    "StressLevel": {
      "type": "object",
      "$ref": "#/definitions/StressLevel"
    },
    "User": {
      "type": "object",
      "$ref": "#/definitions/User"
    }
  },
  "required": [
    "Cards",
    "StressLevel",
    "User"
  ],
  "definitions": {
    "Cards": {
      "type": "object",
      "properties": {
        "Card1": {
          "$ref": "#/definitions/Card"
        },
        "Card2": {
          "$ref": "#/definitions/Card"
        }
      }
    }
  }
}
```

Figura 25. Esquema *json* parte 1.



## 5.7 Storyboard

Neste subcapítulo serão apresentadas as interfaces, de modo, a pré-visualizar como o protótipo do jogo é pretendido.

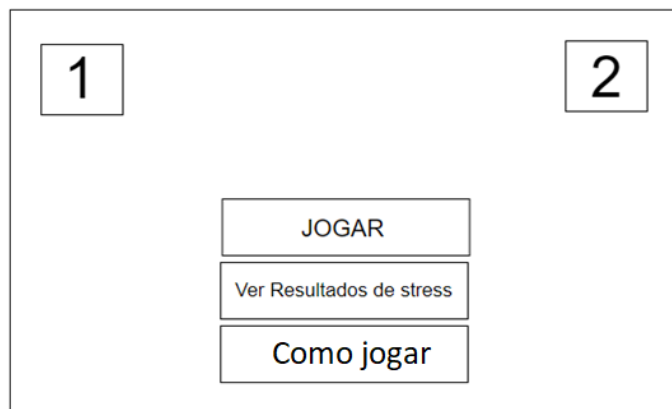


Figura 28. Menu principal.

Na Figura 28, é representado o menu principal, irá possuir cinco botões distintos, nas quais, o número um será um ícone a representar as definições, e o número dois, um ícone a representar o desligar do jogo. Para além desses terá três botões com texto, nas quais o “Jogar”, o “Ver Resultados de Stress” e por último o “Como jogar”.

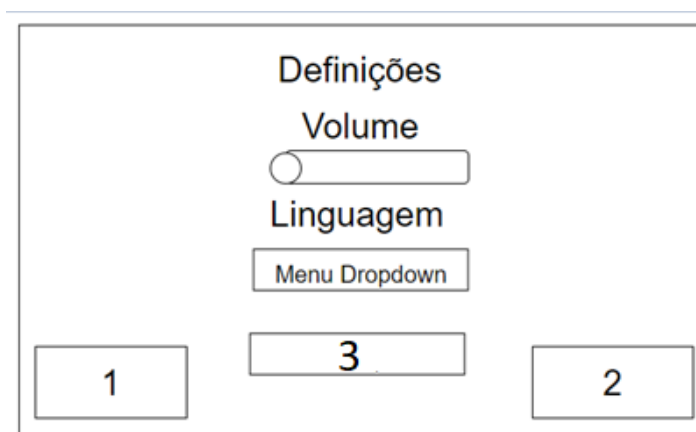


Figura 29. Menu definições

O jogador ao selecionar no ícone 1, da Figura 28, será direcionado para o seguinte menu, ver Figura 29. Neste o jogador poderá aumentar ou diminuir o volume do jogo, escolher o idioma, que irá aparecer no menu *dropdown*, voltar para trás no botão 1, salvar as configurações no botão 2, ou ver informações sobre o jogo (Figura 30), no botão 3.

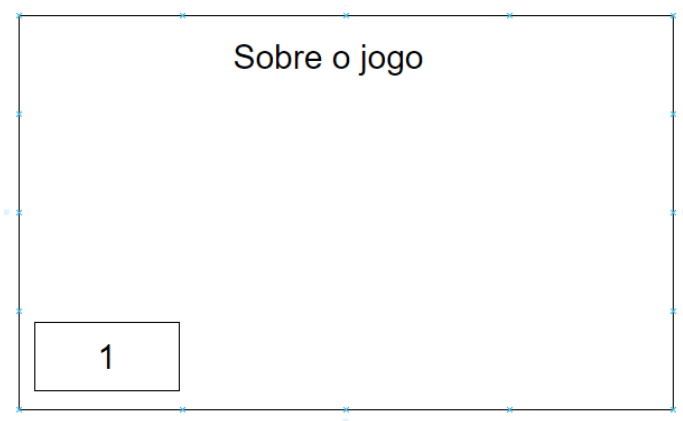


Figura 30. Sobre o jogo.

O jogador, no menu principal se selecionar a opção como jogar, irá ser reencaminhado para uma nova página onde possui o manual de instruções. No botão 1, o jogador irá voltar ao menu definições.

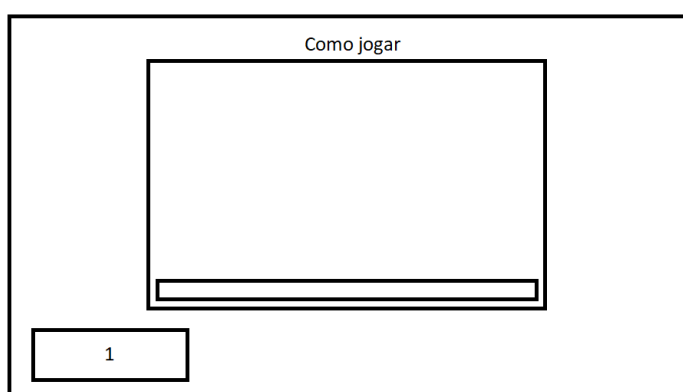


Figura 31. Como jogar.

Na opção "Ver Resultados de Stress", o jogador irá deparar-se com uma página com um gráfico (Figura 32), este serve para mostrar os resultados obtidos do questionário PSS. Este esta na página para onde o botão 2 reencaminha. O botão 1, representa voltar atrás.

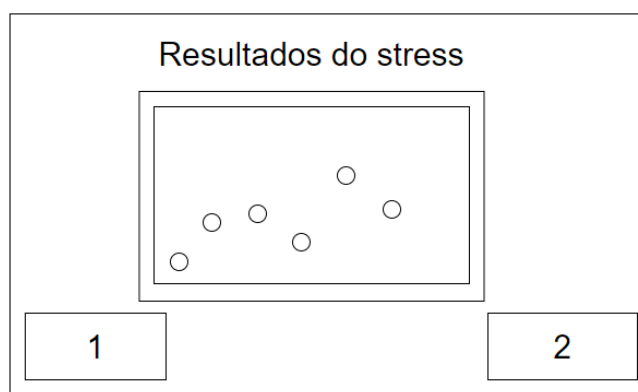
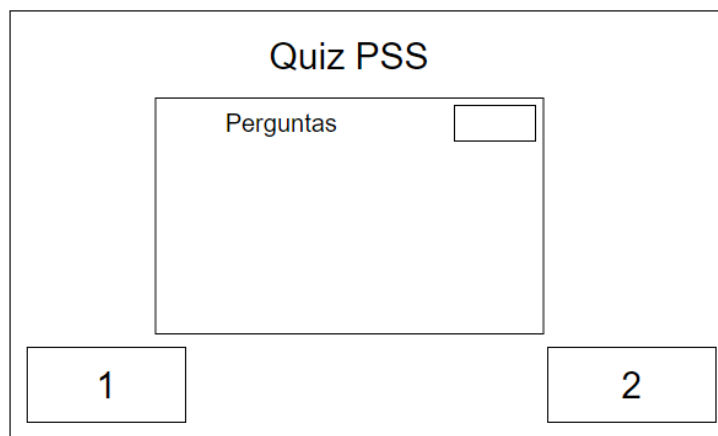


Figura 32. Resultados obtidos do stress.

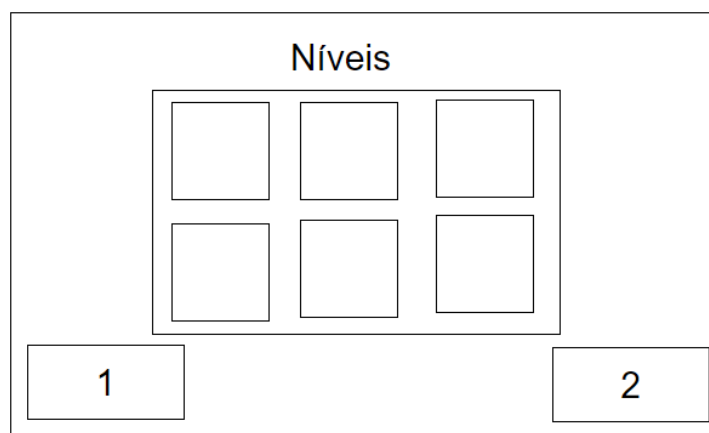
Na Figura 33, representa o questionário falado anteriormente, o jogador poderá selecionar o valor pretendido e submeter o resultado, no botão 2, ou voltar atrás no botão 1.



The image shows a user interface for a 'Quiz PSS'. At the top center, the text 'Quiz PSS' is displayed. Below it is a rectangular area labeled 'Perguntas' (Questions) with a small empty box to its right. At the bottom of the interface, there are two buttons: button '1' on the left and button '2' on the right.

Figura 33. Questionário PSS.

No menu representado na Figura 34, pretende-se mostrar os níveis que o jogador já desbloqueou e poderá selecionar, para jogar. Poderá também ver a coleção de cartas ao selecionar a opção 2, e voltar atrás na opção 1.



The image shows a user interface for a 'Níveis' (Levels) menu. At the top center, the text 'Níveis' is displayed. Below it is a grid of six empty square boxes arranged in two rows and three columns. At the bottom of the interface, there are two buttons: button '1' on the left and button '2' on the right.

Figura 34. Menu de níveis.

Ao selecionar um nível, no menu anterior, o jogador será reencaminhado para o nível selecionado (Figura 35), este consiste num *slide puzzle*. Caso o jogador não queria continuar poderá selecionar a opção 1, que fará com que ele desista do jogo e volte ao menu de níveis. O botão com o ponto de interrogação, irá mostrar o próximo passo ao jogador, sendo um botão de ajuda.

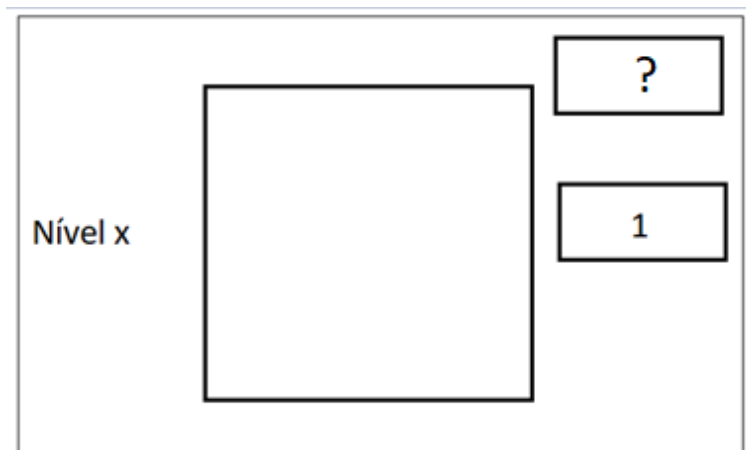


Figura 35. Nível

Na coleção de cartas (Figura 36), o jogador poderá ver a quantidade de cartas que faltas desbloquear e as cartas que já desbloqueou, cada carta irá ter uma determinada informação.

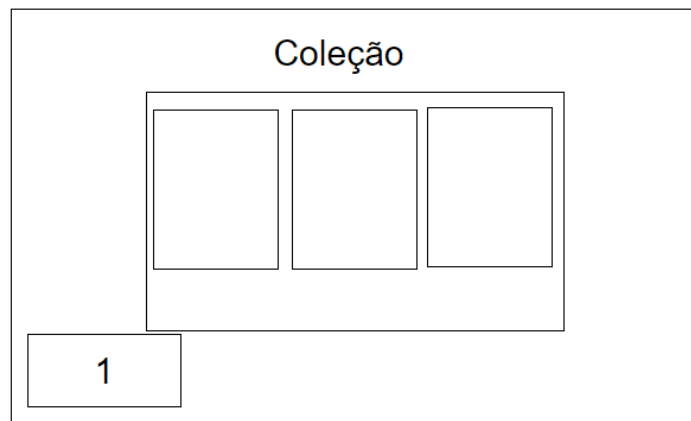


Figura 36. Coleção de cartas.

## 5.8 Paletes de cor a ser utilizada

Através do endereço online <https://colors.co>, a autora criou uma paleta de cores a base do azul (Figura 38), nas quais, de acordo com o artigo disponibilizado pela *Frieze news desk* (DESK, 2019), através de uma pesquisa realizada a 26,596 pessoas pela universidade de Sussex e a empresa G.F Smith, no Reino Unido, concluíram que o azul-escuro é associado com o sentimento de clama. Sendo esta a justificação da escolha da cor inicial para a paleta.



Figura 37. Gradiente da paleta de cores.

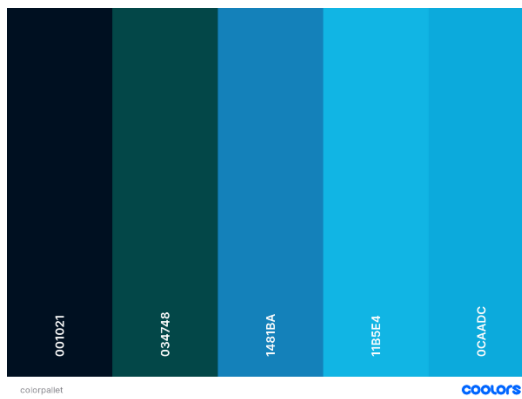


Figura 38. Paleta de cores.

## 6 Construção da Solução

Neste capítulo será descrito detalhadamente o desenvolvimento necessário de modo a obter o protótipo final do jogo sério, ou seja, serão apresentados os desenvolvimentos executados para o cumprimento dos casos de uso referidos no subcapítulo Casos de Uso. Estes serão demonstrados com a imagem do editor *Unity*, e através de segmentos do código em *C#*.

### 6.1 Protótipo Desenvolvido

Primeiro tema a ser descrito sobre o protótipo, será a razão pela qual não foi utilizada base de dados, como estava proposto no Desenho. O *Unity*, fornece um método para salvar os dados de um jogador, as *PlayerPrefs*, como nesta fase do protótipo não se pretendia guardar informação como palavras-passes, dados do utilizador, não foi utilizado a Base de dados.

Para além do descrito anteriormente, a autora, após uma pesquisa exaustiva, não conseguiu encontrar um método de conectar o dispositivo *wearable* que possui, com o *Unity*. Sendo assim, não foi implementado no desenvolvimento a compatibilidade com esse tipo de dispositivos. Em suma, o jogo sério não contém *biofeedback* em relação ao nível de stress do jogador.

De seguida será descrito como a autora implementou e desenvolveu aplicando os tópicos previamente referidos ao longo do documento.

### 6.1.1 Logo do jogo

Utilizando a paleta de cores, a autora criou um logo. Este consiste num comando de consola com um penso, ou seja, juntar o conceito de jogos com o conceito de saúde.



Figura 39. Logo do jogo.

### 6.1.2 Menu principal

O menu principal, Figura 40, possui cinco botões, nas quais, cada um tem um determinado comportamento. O botão no canto superior esquerdo direciona o jogador para as definições, o botão superior direito se selecionado irá fechar o jogo. Os restantes botões irão redirecionar o jogador para o mapa de níveis, o resultado do stress e para uma interface de como jogar, pela ordem respetiva.

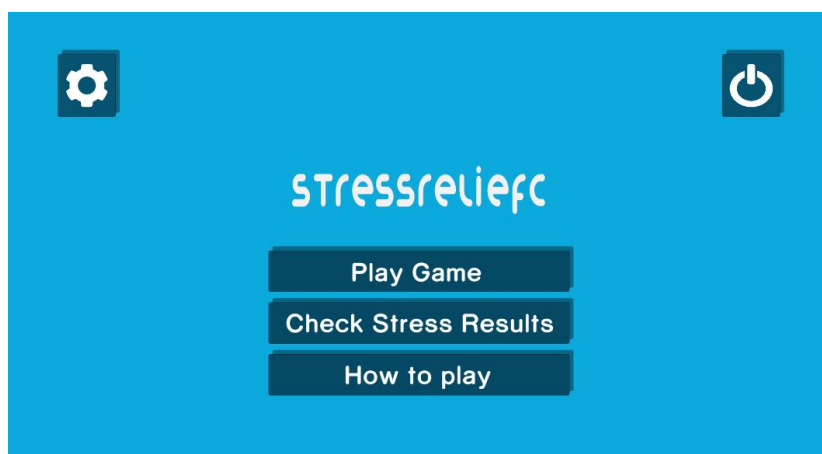


Figura 40. Interface Unity – Menu principal.

O botão “Play Game” têm o comportamento de mudar de cenário, nas quais utilizando o código na Figura 41 e na função *OnClick*, ou seja, quando este for selecionado, adicionar o código C# ao botão (Figura 42).

```

1  using UnityEngine;
2
3  public class OpenScene : MonoBehaviour
4  {
5      public SceneFader fader;
6
7      public void Select(string sceneName)
8      {
9          fader.FadeTo(sceneName);
10     }
11 }
12

```

Figura 41. Código para mudar de cenário com *fader*.

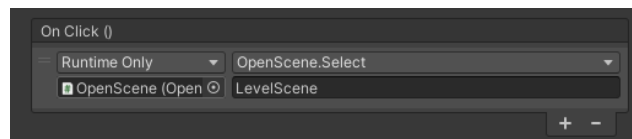


Figura 42. Botão para mudar de cenário.

```

1  using UnityEngine;
2  using UnityEngine.UI;
3  using UnityEngine.SceneManagement;
4  using System.Collections;
5
6  public class SceneFader : MonoBehaviour {
7
8      public Image img;
9      public AnimationCurve curve;
10
11     void Start ()
12     {
13         StartCoroutine(FadeIn());
14     }
15
16     public void FadeTo (string scene)
17     {
18         StartCoroutine(FadeOut(scene));
19     }
20
21     IEnumerator FadeIn ()
22     {
23         float t = 1f;
24
25         while (t > 0f)
26         {
27             t -= Time.deltaTime;
28             float a = curve.Evaluate(t);
29             img.color = new Color(0f, 0f, 0f, a);
30             yield return 0;
31         }
32     }
33
34     IEnumerator FadeOut(string scene)
35     {
36         float t = 0f;
37
38         while (t < 1f)
39         {
40             t += Time.deltaTime;
41             float a = curve.Evaluate(t);
42             img.color = new Color(0f, 0f, 0f, a);
43             yield return 0;
44         }
45
46         SceneManager.LoadScene(scene);
47     }
48 }
49

```

Figura 43. Código do Fader.

O botão “*Check Stress Result*”, possui um comportamento e código parecido com o anterior. Sendo acrescentado código de modo a validar se é a primeira vez que o jogador entre na opção (Figura 44), isto para o gráfico que irá aparecer nos Resultados, ter um valor inicial. Para isso, através do método *PlayerPrefs.HasKey*, irá ser validado se a chave “*FirstTime*” existe, caso não exista, irá abrir o cenário do questionário, caso já exista a chave, irá para os Resultados do Stress.

```

WindowGraph.cs OpenFirstTime.cs x
Miscellaneous Files - OpenFirstT
1 using UnityEngine;
2 using System.Collections;
3
4 public class OpenFirstTime : MonoBehaviour
5 {
6     public SceneFader fader;
7     public GameObject popUp;
8
9     private float wait_time = 3f;
10
11     public void CheckIFitsFristTime(string sceneName)
12     {
13         if (!PlayerPrefs.HasKey("FirstTime"))
14         {
15             PlayerPrefs.SetInt("FirstTime", 0);
16             StartCoroutine(PopUp("PSSQuiz"));
17         }
18         else
19         {
20             fader.FadeTo(sceneName);
21         }
22     }
23
24     IEnumerator PopUp(string sceneName)
25     {
26         if (popUp != null)
27         {
28             popUp.SetActive(true);
29             yield return new WaitForSeconds(wait_time);
30             popUp.SetActive(false);
31             fader.FadeTo(sceneName);
32         }
33     }
34 }
35

```

Figura 44. Código do botão Resultados

O botão de definições, canto superior esquerdo, e o botão “How to Play”, têm ambos o mesmo comportamento, sendo que abrem um painel e não um novo cenário, como se pode verificar no código da Figura 45, é ativado e desativado um painel, nas quais deverá ser referenciado quais na inspeção do elemento (Figura 46), neste caso o botão.

```

Miscellaneous Files - OpenPanel
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.SceneManagement;
5
6
7 public class OpenPanel : MonoBehaviour
8 {
9     public GameObject PanelO;
10    public GameObject PanelH;
11
12    public GameObject PanelTemp;
13
14    float wait_time = 3f;
15
16    public void PanelOpener()
17    {
18        if (PanelH != null && PanelO !=null)
19        {
20            bool isActive = PanelO.isActiveSelf;
21            PanelO.SetActive(!isActive);
22            PanelH.SetActive(isActive);
23        }
24    }
25
26
27    IEnumerator Panel_Open_Await()
28    {
29        if (PanelTemp != null)
30        {
31            PanelTemp.SetActive(true);
32            yield return new WaitForSeconds(wait_time);
33            PanelTemp.SetActive(false);
34            PanelOpener();
35        }
36    }
37
38    public void PanelWaitOpen()
39    {
40        StartCoroutine(Panel_Open_Await());
41    }
42 }

```

Figura 45. Código Abrir Painel.

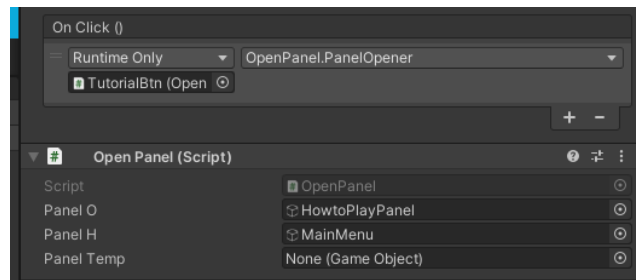


Figura 46. Botão para mudar de painel.

No botão de fechar a aplicação, no canto superior direito, *OnClick* irá executar o código básico (Figura 47) para fechar o jogo.

```

CloseGame.cs  OpenPanel.cs  SceneFader.cs  Open
Miscellaneous Files
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class CloseGame : MonoBehaviour
6  {
7      public void Exit()
8      {
9          Application.Quit();
10     }
11 }
12

```

Figura 47. Código *Quit*.

### 6.1.3 Definições

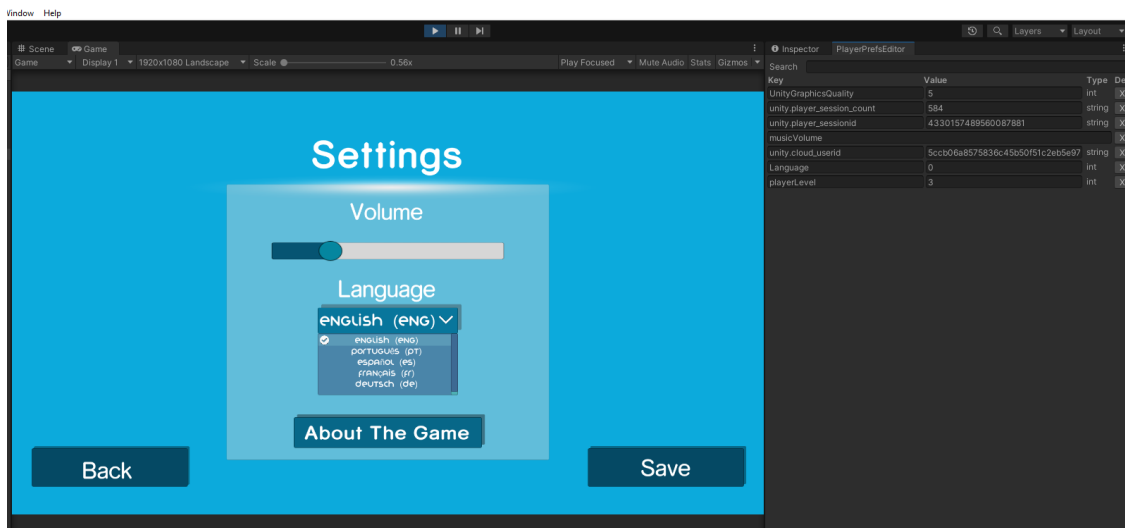
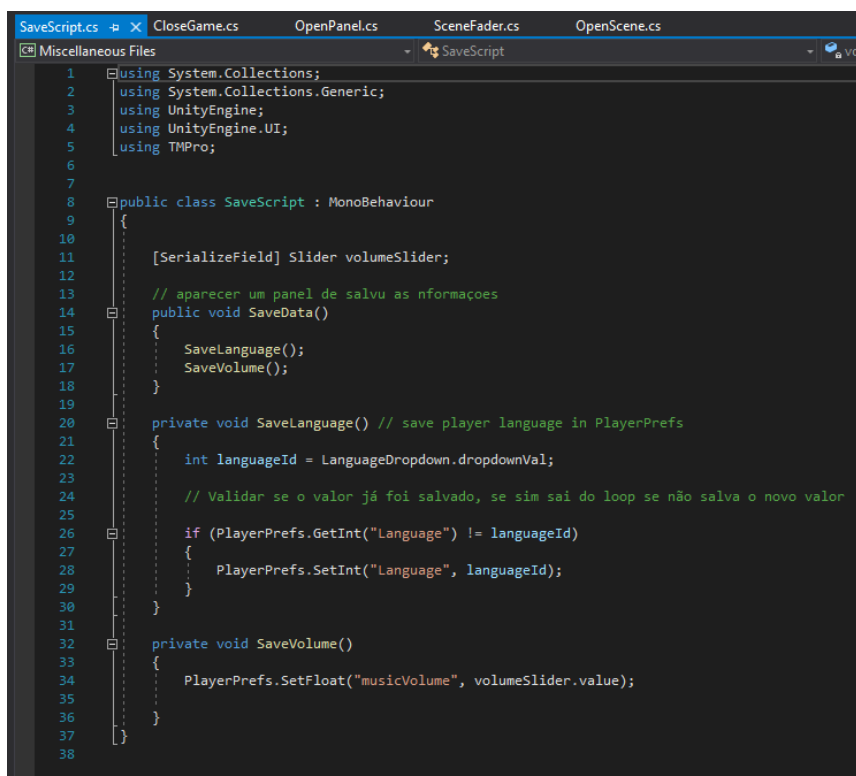


Figura 48. Menu definições – Unity

As definições (Figura 48) é um painel diferente, mas encontra-se no mesmo cenário que o menu principal. Este é responsável por fornecer um método para o jogador configurar o volume do áudio do jogo, mudar o idioma do jogo, e poder ver informações sobre o jogo. Para tal o jogador, executa as mudanças pretendidas e seleciona o botão salvar, em inglês “Save”. Este será responsável por guardar as preferências no *PlayerPrefs*, sendo que o método “Get”, como o nome indica, irá procurar o valor, já o método “Set” irá atribuir o valor (Figura 49).



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5 using TMPro;
6
7
8 public class SaveScript : MonoBehaviour
9 {
10     [SerializeField] Slider volumeSlider;
11
12     // aparecer um panel de salvu as nformações
13     public void SaveData()
14     {
15         SaveLanguage();
16         SaveVolume();
17     }
18
19     private void SaveLanguage() // save player language in PlayerPrefs
20     {
21         int languageId = LanguageDropdown.dropdownVal;
22
23         // Validar se o valor já foi salvado, se sim sai do loop se não salva o novo valor
24
25         if (PlayerPrefs.GetInt("Language") != languageId)
26         {
27             PlayerPrefs.SetInt("Language", languageId);
28         }
29     }
30
31     private void SaveVolume()
32     {
33         PlayerPrefs.SetFloat("musicVolume", volumeSlider.value);
34     }
35
36
37 }
38
```

Figura 49. Código para guardar as preferências do jogador.

O “*Dropdown Menu*” implementado, contém uma solução de modo a mal o jogador selecione uma opção, todas as palavras irão automaticamente alterar para linguagem escolhida. O idioma só irá alterar no menu presente, nos restantes irá alterar no caso de o botão salvar ser pressionado. Para isso, foi gerado um ficheiro de texto guardado como csv (*Comma-separated values*), que contém as traduções, como se pode verificar na Figura 50. O ficheiro contém uma primeira linha, que guarda o ID, ou seja, a chave pela qual o código irá encontrar o que deverá substituir, e pelos diversos idiomas, neste caso foram escolhidos cinco.

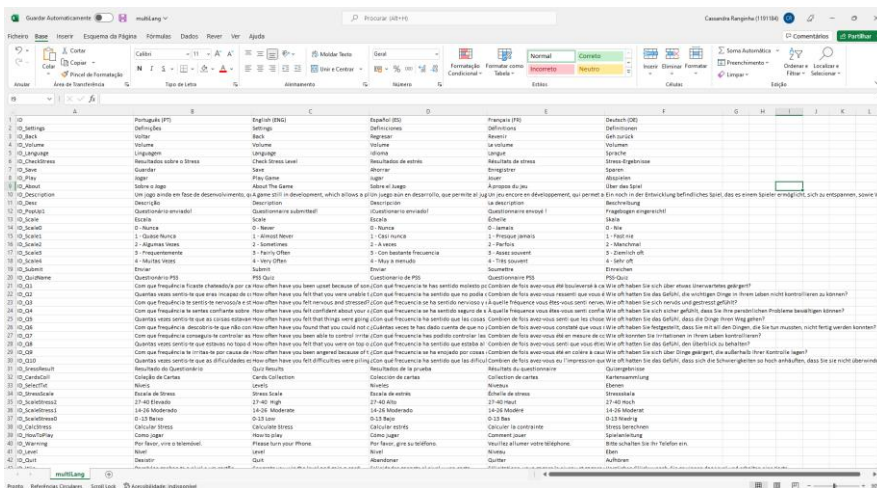


Figura 50. Ficheiro multilang.

O ficheiro é lido pelo código representado na Figura 51, este adiciona os diversos idiomas a uma lista (método GetAvailableLanguage), tal como adiciona a tradução para cada chave num dicionário (método GetTextFromId).

```

CVSParser.cs LanguageDropdown.cs LocalizationTexts
Miscellaneous Files
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System.Linq;

public class CVSParser : MonoBehaviour
{
    static private List<string> languageList = new List<string>();
    static private Dictionary<string, List<string>> languageDictionary = new Dictionary<string, List<string>>();
    // save all the words for a specific language

    static public string[] SplitLine (string line)
    {
        return
            (from System.Text.RegularExpressions.Match m in System.Text.RegularExpressions.Regex.Matches(line,
                @"^(((\w+)(\s+))*)((\w+)(\s+))*)((\w+)(\s+))*)((\w+)(\s+))*)$")
            .Select(m.Groups[1].Value).ToArray();
    }

    // Will return all the available languages in the excel file
    static public List<string> GetAvailableLanguage()
    {
        if (languageList.Count == 0)
        {
            var csvFile = Resources.Load<TextAsset>("multilang");
            string[] lines = csvFile.text.Split("\n");
            languageList = new List<string>(SplitLine(lines[0]));
            languageList.RemoveAt(0);
        }

        return languageList;
    }

    static public string GetTextFromId(string Id, int languageIndex)
    {
        if (languageDictionary.Count == 0)
        {
            var csvFile = Resources.Load<TextAsset>("multilang");
            string[] lines = csvFile.text.Split("\n");

            for (int i = 1; i < lines.Length; i++)
            {
                string[] row = SplitLine(lines[i]);

                if (row.Length > 1)
                {
                    List<string> words = new List<string>(row);
                    words.RemoveAt(0);
                    languageDictionary.Add(row[0], words);
                }
            }

            var values = languageDictionary[Id];
            return values[languageIndex];
        }
    }
}

```

Figura 51. Código de leitura dos idiomas.

No objeto “dropdown”, adicionou-se o código responsável por validar quando há uma mudança de valor selecionado (Figura 52), tal como atribui as opções de idioma às várias linhas do menu (Figura 53).

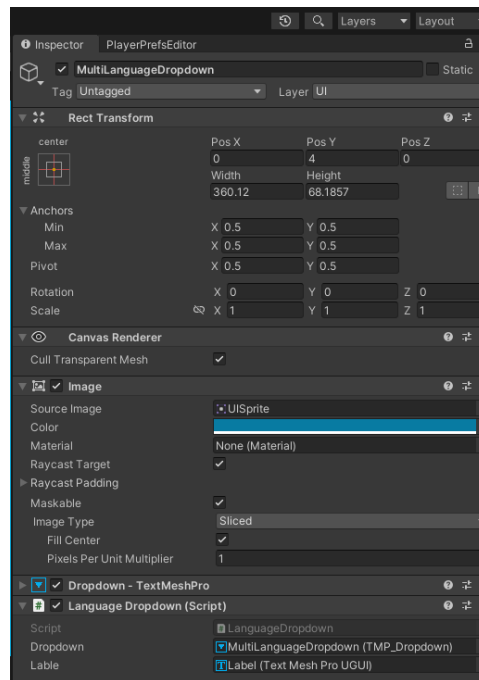


Figura 52. Código associado ao objeto.

```

LanguageDropdowns - LocalizationText.cs
Miscellaneous Files - LanguageDropdown
1 using System;
2 using System.Collections;
3 using System.Collections.Generic;
4 using UnityEngine;
5 using UnityEngine.UI;
6 using TMPro;
7
8 [RequireComponent(typeof(TMP_Dropdown))]
9 public class LanguageDropdown : MonoBehaviour
10 {
11     static public Action<int> ChangeLanguage;
12     public TMP_Dropdown dropdown;
13     public TMP_Text label;
14
15     public static int dropdownVal;
16
17     public void LanguageChanged()
18     {
19         if (ChangeLanguage != null)
20             ChangeLanguage(dropdown.value);
21
22         dropdown.captionText.text = CVSParser.GetAvailableLanguage()[dropdown.value];
23         label.text = dropdown.captionText.text;
24
25         dropdownVal = dropdown.value;
26
27     }
28
29     void Start()
30     {
31         PopulateDropdown();
32     }
33
34     void PopulateDropdown()
35     {
36         dropdown.ClearOptions();
37         dropdown.AddOptions(CVSParser.GetAvailableLanguage());
38         LanguageChanged();
39     }
40 }

```

Figura 53. Código do *Dropdown*.

No caso dos objetos de texto, foi necessário adicionar a cada um, o código “*LocalizationText*” nas propriedades (Figura 54). Este é responsável por procurar as preferências do jogador, e atribuir o valor da tradução, de acordo com a chave passada no código (Figura 55).

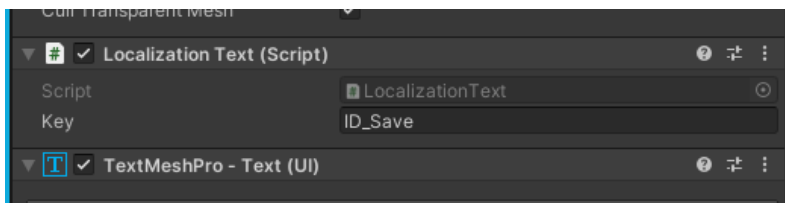


Figura 54. Propriedades do Texto.

```

1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using UnityEngine.UI;
6  using TMPro;
7
8  [RequireComponent(typeof(TMP_Text))]
9  public class LocalizationText : MonoBehaviour
10 {
11     public string key;
12
13     public void ChangeLanguage(int index)
14     {
15         gameObject.GetComponent<TMP_Text>().text = CVSParser.GetTextFromId(key, index);
16     }
17
18     void Start()
19     {
20         int languageId = PlayerPrefs.GetInt("Language");
21         ChangeLanguage(languageId);
22     }
23
24     private void OnEnable()
25     {
26         LanguageDropdown.ChangeLanguage += ChangeLanguage;
27     }
28
29     private void OnDisable()
30     {
31         LanguageDropdown.ChangeLanguage -= ChangeLanguage;
32     }
33
34 }
35
36

```

Figura 55. Código para mudar o texto conforme idioma.

Em relação à alteração do volume foi desenvolvido o código descrito na Figura 56, e atribuído ao *slider* do volume (Figura 57).

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5
6  public class SoundManager : MonoBehaviour
7  {
8      [SerializeField] Slider volumeSlider;
9
10     // Start is called before the first frame update
11     void Start()
12     {
13         if (!PlayerPrefs.HasKey("musicVolume"))
14         {
15             PlayerPrefs.SetFloat("musicVolume", 1);
16             Load();
17         }
18         else
19         {
20             Load();
21         }
22     }
23
24     public void ChangeVolume()
25     {
26         Audiolistener.volume = volumeSlider.value;
27         //Save();
28     }
29
30     private void Load()
31     {
32         volumeSlider.value = PlayerPrefs.GetFloat("musicVolume");
33     }
34 }
35
36

```

Figura 56. Código do áudio.

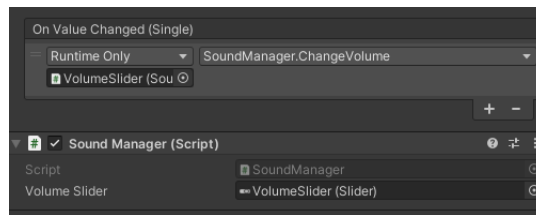


Figura 57. Preferências do *Slider*.

#### 6.1.4 Questionário PSS

Este é o cenário que irá permitir ao jogador realizar e submeter o questionário PSS, já descrito no subcapítulo Perceived Stress Scale. Para tal a interface é composta por 10 objetos nas quais representam o valor atribuí a cada questão, este valor será o selecionado no *dropdown* Figura 58.

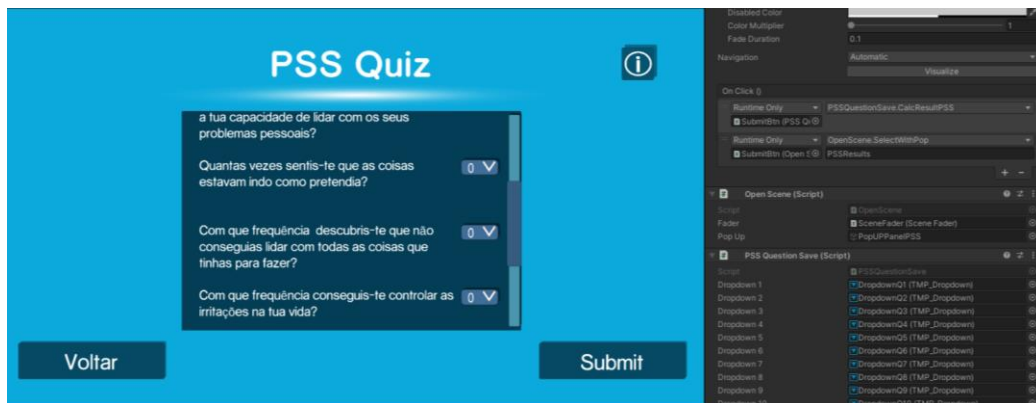


Figura 58. Questionário PSS.

O jogador quando seleciona o botão “Submit”, irá ser direcionado para os Resultados do questionário. Será neste, a qual é associado o código responsável pelo cálculo final, representado na Figura 59. Primeiro passo do código é executar o cálculo direto, obtido nas questões número 1,2,3,6,9 e 10, segundo passo será calcular o valor entre o resultado, duas a duas, das questões 4,5,7 e 8, para tal, criado o método *ReverseValue()*, no final é executado o somatório de todos os valores obtidos.

```

7
8 public class PSSQuestionSave : MonoBehaviour
9
10 {
11     public TMP_Dropdown dropdown1, dropdown2, dropdown3, dropdown4, dropdown5, dropdown6, dropdown7, dropdown8, dropdown9, dropdown10;
12     int calcf1; // irá calcular o valor da soma das respostas as seguintes perguntas 1,2,3,6,9,10
13     int resultCalA, resultCalc8; // reposavel por guardar o valor sumado, dois a dois do contrário das questões 4,5,7,8
14     int calTotal;
15
16     string time;
17
18     [System.Serializable]
19     public class Result
20     {
21         public string datetime { get; set; }
22         public string resultcal { get; set; }
23     }
24
25     public void CalcResultPSS() // será chamada pelo botão submit
26     {
27         calcf1 = dropdown1.value + dropdown2.value + dropdown3.value + dropdown6.value + dropdown9.value + dropdown10.value;
28         ReverseValue();
29         calTotal = calcf1 + resultCalc8 + resultCalA;
30
31         //ReadCSVfile();
32         AppendCSVfile(calTotal);
33     }
34
35     private void ReverseValue()
36     {
37
38         switch (dropdown4.value, dropdown5.value)
39         {
40             /* como os valores invertem sendo
41              * 0 = 4;
42              * 1 = 3;
43              * 2 = 2;
44              * 3 = 1;
45              * 4 = 0
46              */
47             case (0, 0):
48                 resultCalA = 8;
49                 break;
50             case (1, 0):
51                 resultCalA = 7;
52                 break;
53             case (0, 2):
54                 resultCalA = 6;
55                 break;
56             case (2, 0):
57                 resultCalA = 5;
58                 break;
59             case (0, 3):
60                 resultCalA = 4;
61                 break;
62             case (3, 0):
63                 resultCalA = 3;
64                 break;
65             case (0, 4):
66                 resultCalA = 2;
67                 break;
68             case (4, 0):
69                 resultCalA = 1;
70                 break;
71             case (2, 2):
72                 resultCalA = 1; // 1 + 0 ou 0 + 1 = 1
73                 break;
74             case (1, 4):
75                 resultCalA = 0; // 0 + 0 = 0
76                 break;
77             case (4, 4):
78                 resultCalA = 0; // 0 + 0 = 0
79                 break;
80             case (3, 3):
81                 resultCalA = 0; // 0 + 0 = 0
82                 break;
83             case (4, 3):
84                 resultCalA = 0; // 0 + 0 = 0
85                 break;
86             case (4, 2):
87                 resultCalA = 0; // 0 + 0 = 0
88                 break;
89             case (4, 1):
90                 resultCalA = 0; // 0 + 0 = 0
91                 break;
92             case (4, 0):
93                 resultCalA = 0; // 0 + 0 = 0
94                 break;
95         }
96     }
97 }

```

Figura 59. Código do cálculo do PSS.

O valor total calculado nos métodos anteriores, será guardado num ficheiro csv, de modo, a ser utilizado no cenário Resultado do Questionário, juntamente com a dada realizada, como se pode verificar no código da Figura 60. Na Figura 61 segue-se um exemplo de como os dados são guardados.

```

// função serve para guardar os dados num csv, os dados do questionario PSS
static void AppendCSVfile(int calTotal)
{
    string time = System.DateTime.UtcNow.ToLocalTime().ToString("yyyy/MM/dd HH:mm:ss");
    string filename = Application.dataPath + "/Resources/resultStress.csv";

    //using append = true
    using (StreamWriter sr = new StreamWriter(filename, true))
    {
        sr.WriteLine(time + "," + calTotal);
    }
}

```

Figura 60. Código que salva no csv.

	A	B	C	D	E	F
1	12/12/1990 00:00	0				
2	13/10/2022 12:56	16				
3	13/10/2022 12:56	18				

Figura 61. Ficheiro csv de resultados.

### 6.1.5 Resultado do Questionário

Após o jogador responder ao questionário, descrito anteriormente, é necessário apresentar os resultados, para tal, desenvolveu-se a seguinte interface, representada na Figura 62.

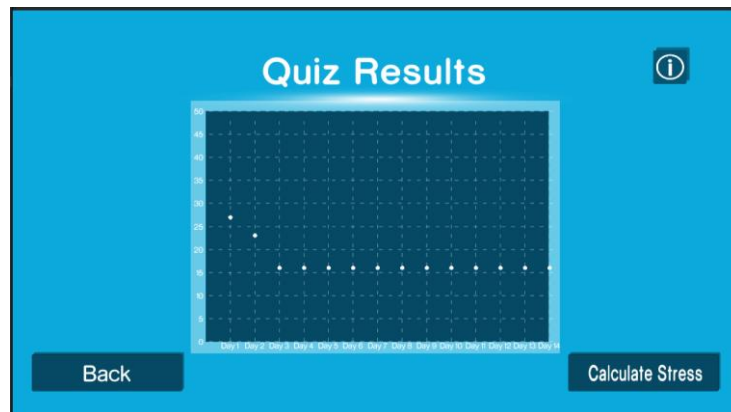


Figura 62. Resultado do questionário.

Foi desenvolvido o código descrito na Figura 64 e na Figura 64, para desenhar o gráfico. O código começa por obter os dados do questionário salvados no ficheiro csv (Figura 64) e finaliza a chamar o método responsável pela construção do gráfico.

```

9 public class WindowGraph : MonoBehaviour
10 {
11     [SerializeField]
12     private Sprite circleSprite;
13     private RectTransform graphContainer;
14     private RectTransform labelTemplateX, labelTemplateY, dashTemplateX, dashTemplateY;
15
16     private List<int> valueList;
17
18     private void Awake()
19     {
20         graphContainer = transform.Find("graphContainer").GetComponent<RectTransform>();
21
22         labelTemplateX = graphContainer.Find("labelTemplateX").GetComponent<RectTransform>();
23         labelTemplateY = graphContainer.Find("labelTemplateY").GetComponent<RectTransform>();
24
25         dashTemplateX = graphContainer.Find("dashTemplateX").GetComponent<RectTransform>();
26         dashTemplateY = graphContainer.Find("dashTemplateY").GetComponent<RectTransform>();
27
28         ReadCSVSSFile();
29     }
30
31     private void ReadCSVSSFile()
32     {
33         string filename = Application.dataPath + "/Resources/resultstress.csv";
34         valueList = new List<int>();
35
36         using (var reader = new StreamReader(filename))
37         {
38             while (!reader.EndOfStream)
39             {
40                 var line = reader.ReadLine();
41                 var values = line.Split(",");
42
43                 valueList.Add(int.Parse(values[1]));
44             }
45
46             ShowGraph(valueList);
47         }
48     }
49
50 }

```

Figura 63. Código desenhar gráfico – Parte 1.

O método *ShowGraph()*, é o responsável pelo desenho, tanto dos eixos, como dos pontos que são atribuídos aos valores. Para tal é necessário um ciclo de modo a percorrer a lista de dados guardados e desenhar os pontos, utilizando o método *CreateCircle()*.

```

52 private GameObject CreateCircle(Vector2 anchoredPosition)
53 {
54     GameObject gameObject = new GameObject("circle", typeof(Image));
55     gameObject.transform.SetParent(graphContainer, false);
56     gameObject.GetComponent<Image>().sprite = circleSprite;
57     RectTransform rectTransform = gameObject.GetComponent<RectTransform>();
58     rectTransform.anchoredPosition = anchoredPosition;
59     rectTransform.sizeDelta = new Vector2(11, 11);
60     rectTransform.anchorMin = new Vector2(0, 0);
61     rectTransform.anchorMax = new Vector2(0, 0);
62
63     return gameObject;
64 }
65
66 private void ShowGraph(List<int> valueList)
67 {
68     float graphHeight = graphContainer.sizeDelta.y;
69     float yMaximum = 50f;
70     float xSize = 65f;
71
72     //GameObject lastCircleGameObject = null;
73
74     for (int i = 1; i < valueList.Count; i++)
75     {
76         //float xPosition = xSize + i * xSize;
77         float xPosition = i * xSize;
78         float yPosition = (valueList[i] / yMaximum) * graphHeight;
79         GameObject circleGameObject = CreateCircle(new Vector2(xPosition, yPosition));
80
81         RectTransform labelX = Instantiate(labelTemplateX);
82         labelX.SetParent(graphContainer);
83         labelX.gameObject.SetActive(true);
84         labelX.anchoredPosition = new Vector2(xPosition, -10f);
85         labelX.GetComponent<Text>().text = "Day " + i.ToString();
86
87         RectTransform dashX = Instantiate(dashTemplateX);
88         dashX.SetParent(graphContainer);
89         dashX.gameObject.SetActive(true);
90         dashX.anchoredPosition = new Vector2(xPosition, -10f);
91     }
92
93     int separatorCount = 10;
94     for (int i = 0; i <= separatorCount; i++) {
95         RectTransform labelY = Instantiate(labelTemplateY);
96         labelY.SetParent(graphContainer, false);
97         labelY.gameObject.SetActive(true);
98         float normalizedValue = i * 1f / separatorCount;
99         labelY.anchoredPosition = new Vector2(-7f, normalizedValue * graphHeight);
100         labelY.GetComponent<Text>().text = Mathf.RoundToInt(normalizedValue * yMaximum).ToString();
101
102         RectTransform dashY = Instantiate(dashTemplateY);
103         dashY.SetParent(graphContainer);
104         dashY.gameObject.SetActive(true);
105         dashY.anchoredPosition = new Vector2(0f, normalizedValue * graphHeight);
106     }
107 }
108
109 }
110

```

Figura 64. Código desenhar gráfico – Parte 2.

### 6.1.6 Sobre o jogo

Sobre o jogo, Figura 65, é um painel básico, que contém as informações sobre o jogo. É composto por uma pequena descrição, o logo do jogo, o ano de desenvolvimento e o número da versão.



Figura 65. Sobre o jogo.

### 6.1.7 Seleção de níveis

No cenário, seleção de níveis (Figura 66), é responsável por disponibilizar os níveis que o jogador já desbloqueou e método de navegação para mostrar a coleção de cartas do jogador. Para desbloquear os níveis, foi necessário adicionar todos os objetos que correspondem aos vários níveis nas propriedades do código (Figura 67), assim como obter o nível do jogador através do método *PlayerPrefs.GetInt()*, e executar um ciclo, de modo a desativar todos os níveis disponíveis e a ativar os botões até ao nível seguinte que o jogador estava (Figura 68). Por exemplo, caso o jogador tenha vencido o nível 3, o nível 4 já irá desbloquear.

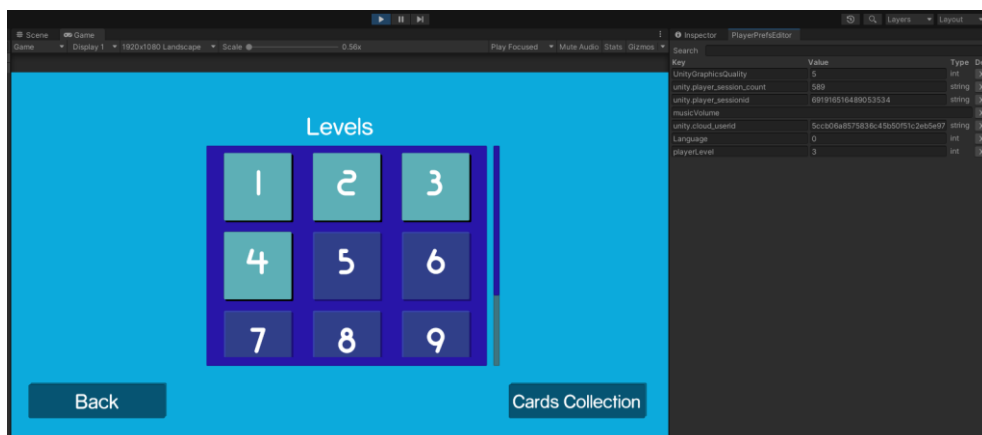


Figura 66. Níveis bloqueados.

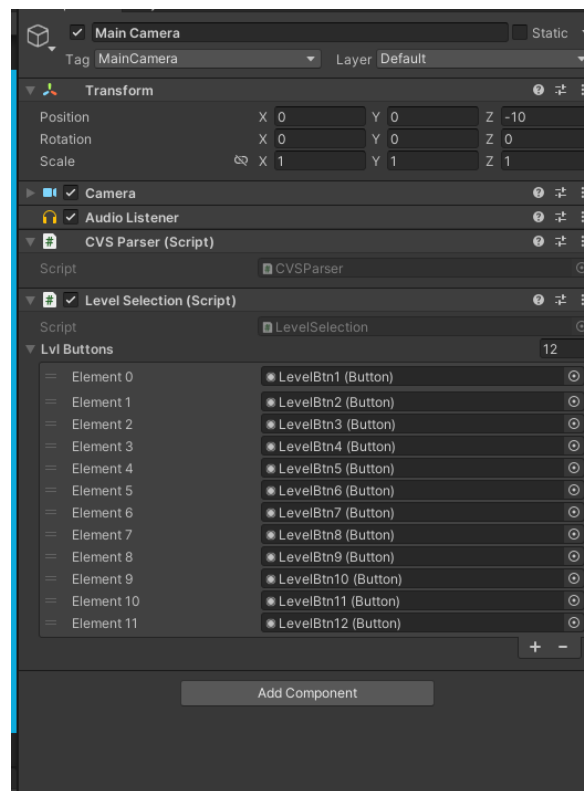


Figura 67. Número de níveis disponíveis.

```

LevelSelection.cs*
Miscellaneous Files
LevelSelection
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5
6 public class LevelSelection : MonoBehaviour
7 {
8     public Button[] lvlButtons;
9     // Start is called before the first frame update
10    void Start()
11    {
12        int levelAt = PlayerPrefs.GetInt("playerLevel", 0);
13        Debug.Log(lvlButtons.Length);
14        // percorrer a lista de níveis que existe
15        for (int i = 0; i < lvlButtons.Length ; i++)
16        {
17            lvlButtons[i].interactable = false;
18            if (i <= levelAt)
19            {
20                lvlButtons[i].interactable = true;
21            }
22        }
23    }
24 }
25
26

```

Figura 68. Código de bloqueio dos níveis.

### 6.1.8 Carta e Coleção de cartas

Quando o jogador seleciona a opção “Cards Collection”, representada na Figura 66, este será direcionado para o cenário que irá mostrar as cartas que o jogador possui (Figura 69).

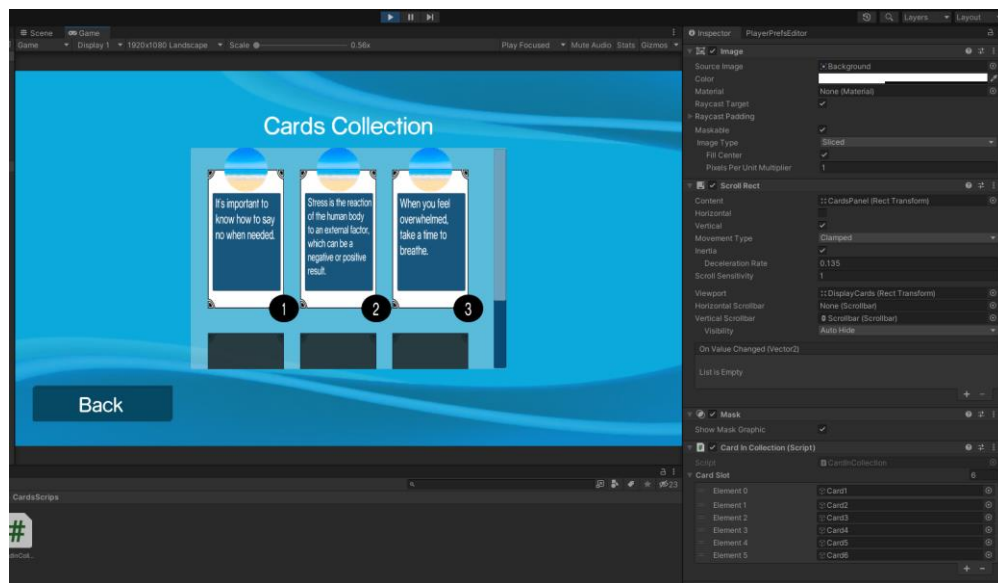


Figura 69. Coleção de Cartas – Unity.

O primeiro passo para a criação desta coleção, foi criado uma Classe de cartas (Figura 70), nas quais irá guardar a imagem, a descrição e o nível à qual a carta corresponde. Após isso foi desenvolvido um *script*, responsável por ler um ficheiro csv (Figura 71), onde estarão guardados os dados de cada carta (Figura 72). No final para a coleção ficar funcional foram atribuídos os vários objetos ao código de *Unity* (Figura 73), e criada uma função de modo a bloquear as cartas cujo jogador ainda não desbloqueou ao ganhar o nível correspondente (Figura 74).

```

Card.cs  x  LevelSelection.cs*
Miscellaneous Files  Card
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  [System.Serializable]
6  public class Card : MonoBehaviour
7  {
8      //[TextArea(1,3)]
9      public string cardLevel;
10     public string cardDes;
11
12     public Sprite thisImage;
13
14     public Card()
15     {
16     }
17
18     public Card(string CardLevel, string CardDes, Sprite ThisImage)
19     {
20     }
21     {
22         cardLevel = CardLevel;
23         cardDes = CardDes;
24         thisImage = ThisImage;
25     }
26
27

```

Figura 70. Classe Carta.

```

public class CardInCollection : MonoBehaviour
{
    private List<Card> thisCard = new List<Card>();
    public GameObject[] cardSlot;

    // Start is called before the first frame update
    void Start()
    {
        //thisCard[0] = CardDatabase.cardList[thiscardLevel];
        ReadCSVCards();
    }

    /**for (int i = 1; i < Playerlevel; i++){
     * carrega todas as cartas do jogador
     */
    private void ReadCSVCards()
    {
        string filename = Application.dataPath + "/Resources/cardsInfo.csv";

        using (var reader = new StreamReader(filename))
        {
            while (!reader.EndOfStream)
            {
                var line = reader.ReadLine();
                var valuesCards = line.Split(";");
                thisCard.Add(new Card(valuesCards[0], valuesCards[1], Resources.Load<Sprite>(valuesCards[2].ToString()) ));
            }
        }
        DisplayCards();
    }
}

```

Figura 71. Código para ler o ficheiro – Coleção de cartas.

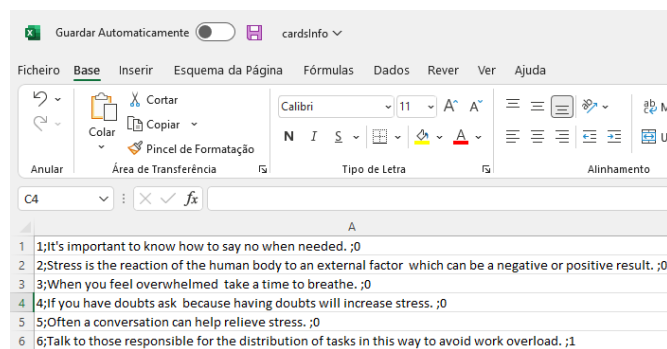


Figura 72. Ficheiro coleção cartas.

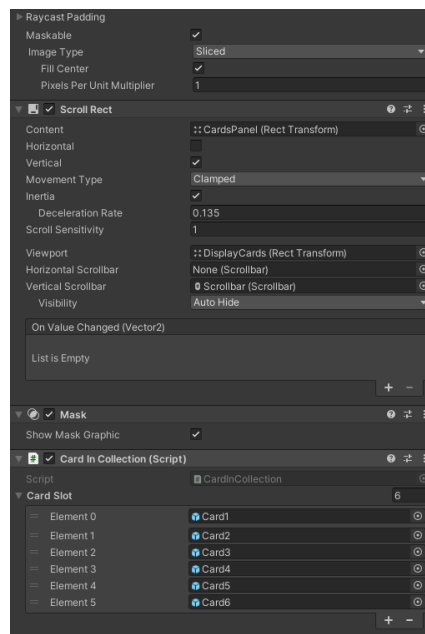


Figura 73. Atribuição dos objetos.

```
private void DisplayCards()
{
    int playerLevel = PlayerPrefs.GetInt("playerLevel");
    for (int i = 0; i < thisCard.Count; i++)
    {
        cardslot[i].transform.GetChild(0).transform.GetChild(0).GetComponent<Image>().sprite = thisCard[i].thisImage;
        cardslot[i].transform.GetChild(0).transform.GetChild(1).transform.GetChild(0).GetComponent<TMP_Text>().text = thisCard[i].cardLevel;
        cardslot[i].transform.GetChild(0).transform.GetChild(2).transform.GetChild(0).GetComponent<TMP_Text>().text = thisCard[i].cardDes;

        // mudar a cor e transparencia das cartas que o jogador não possui.
        cardslot[i].transform.GetChild(0).GetComponent<Image>().color = new Color32(25, 25, 25, 202);
        cardslot[i].transform.GetChild(0).transform.GetChild(1).GetComponent<Image>().color = new Color32(0, 0, 0, 0); //Leveling
        cardslot[i].transform.GetChild(0).transform.GetChild(1).transform.GetChild(0).GetComponent<TMP_Text>().color = new Color32(0, 0, 0, 0); //LevelText
        cardslot[i].transform.GetChild(0).transform.GetChild(0).GetComponent<Image>().color = new Color32(0, 0, 0, 0); //CardFrameMask
        cardslot[i].transform.GetChild(0).transform.GetChild(2).GetComponent<Image>().color = new Color32(0, 0, 0, 0); //CardDesing
        cardslot[i].transform.GetChild(0).transform.GetChild(2).transform.GetChild(0).GetComponent<TMP_Text>().color = new Color32(0, 0, 0, 0); //Desciptxt

        for (int j = 0; j < playerLevel; j++)
        {
            cardslot[j].transform.GetChild(0).GetComponent<Image>().color = new Color32(255, 255, 255, 255); // CardImage
            cardslot[j].transform.GetChild(0).transform.GetChild(1).GetComponent<Image>().color = new Color32(0, 0, 0, 255); //Leveling
            cardslot[j].transform.GetChild(0).transform.GetChild(1).transform.GetChild(0).GetComponent<TMP_Text>().color = new Color32(255, 255, 255, 255); //LevelText
            cardslot[j].transform.GetChild(0).transform.GetChild(0).GetComponent<Image>().color = new Color32(29, 182, 149, 255); //CardFrameMask
            cardslot[j].transform.GetChild(0).transform.GetChild(2).GetComponent<Image>().color = new Color32(29, 182, 149, 255); //CardDesing
            cardslot[j].transform.GetChild(0).transform.GetChild(2).transform.GetChild(0).GetComponent<TMP_Text>().color = new Color32(255, 255, 255, 255); //Desciptxt
        }
    }
}
```

Figura 74. Bloquear cartas ainda não adquiridas.

### 6.1.9 Níveis

Para os níveis foi utilizado um *asset* do *Unity*, chamado SlidingTilePuzzle (Games, 2016). Este foi adicionado (Figura 75) e alterado (Figura 76) para corresponder ao contexto do projeto.



Figura 75. Jogo de Puzzle.

```
public void AfterComplete()
{
    //validar se o player pref é maior que o que esta a ser chamado (o jogador pode querer repetir níveis- para mais tarde ganhar mal
    if (LevelNumber > PlayerPrefs.GetInt("playerLevel", 0))
    {
        PlayerPrefs.SetInt("playerLevel", LevelNumber);

        StartCoroutine(WaitForTime(sceneName));
    }
}

IEnumerator WaitForTime(string sceneName)
{
    float wait_time = 3f;
    if (PanelSucess is not null)
    {
        PanelSucess.SetActive(true);
        yield return new WaitForSeconds(wait_time);
        PanelSucess.SetActive(false);
        Fader.FadeTo(sceneName);
    }
}
}
```

Figura 76. Código Adicionado ao asset.

### 6.1.10 Como jogar

Neste cenário é utilizado um *slider* vertical, de modo a verificar todas as instruções do jogo (Figura 77).

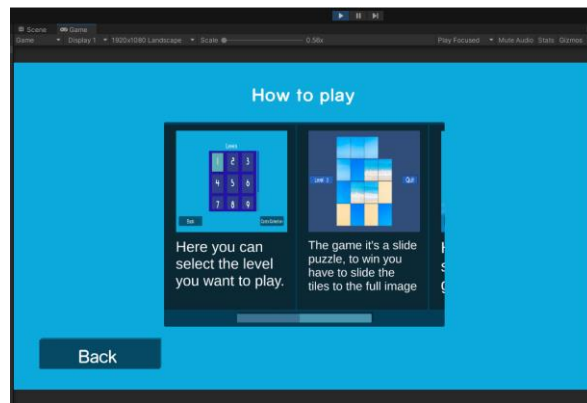


Figura 77. Como jogar – Manual.

## 7 Experimentação e avaliação

Neste capítulo será apresentada a metodologia de avaliação que irá ser utilizada. Nas quais, avaliação de qualidade, utilizando a ferramenta *Quantitative Evaluation Framework* e utilizando um questionário baseado no *System Usability Scale*.

Será essencial adquirir *feedback* das partes interessadas, sendo estes os colaboradores de um ambiente cooperativo e pessoas interessadas em conhecer mais sobre o *stress*. Este será recolhido através da realização de um questionário.

### 7.1 Hipótese de investigação

O projeto a ser desenvolvido tem como objetivo aliviar o stress dos colaboradores em ambiente cooperativo, tal como, sensibilizar e fornecer informações sobre este tema utilizando um jogo sério. De modo a verificar que o projeto desenvolvido atinge o seu

objetivo, ou seja, aceitar ou rejeitar o projeto, será realizado um teste de hipóteses. Este teste consiste na avaliação de dois possíveis resultados, a hipótese nula (H0) e a hipótese alternativa (H1) segue-se na **Erro! A origem da referência não foi encontrada.** uma descrição de cada uma.

Tabela 33. Tabela de Hipóteses

<b>Hipótese Nula (H0)</b>	O jogo não contribui para o alívio do stress dos jogadores nem com o fornecimento de informações sobre este tema.
<b>Hipótese Alternativa (H1)</b>	O jogo contribui para o alívio do stress dos jogadores e com o fornecimento de informações sobre este tema.

## 7.2 Metodologia de avaliação

Para avaliar o projeto desenvolvido e validar qual hipótese de investigação em que o protótipo se adequa, será utilizada uma metodologia baseada em avaliar a qualidade do produto, nas quais será utilizado a ferramenta *Quantitative Evaluation Framework (QEF)*, e será utilizado um questionário, baseado no *System Usability Scale (SUS)*. Este será apresentado a um grupo de indivíduos, de modo a obter a opinião de restantes utilizadores. Estes métodos serão descritos nos subcapítulos seguintes.

## 7.3 Indicadores e fontes de informação

Os indicadores e fontes de informação, consistiu no fornecimento do projeto desenvolvido a um grupo de 10 colaboradores, e obter o *feedback* destes. Este foi através de um questionário que será descrito no subcapítulo Questionário. Para além disso foi recorrido ao QEF, anteriormente falado, para avaliar a qualidade do projeto, tendo em consideração que alguns pontos do QEF, foram avaliados de acordo com o resultado do questionário realizado. Os restantes requisitos foram autoavaliados pela autora da dissertação e classificados conforme as métricas definidas (Ver Anexo A).

## 7.4 Grandezas de avaliação

De modo a executar uma avaliação ao produto final, foram selecionadas três grandezas, nas quais, Pedagógica, Ergonómica e Técnica. Na Tabela 34 segue-se a descrição de cada grandeza selecionada.

Tabela 34. Grandezas de avaliação

<b>Pedagógica</b>	Grandeza que irá permitir avaliar se o jogo sério desenvolvido teve um efeito positivo em relação à aprendizagem do jogador. Neste caso será adquirir conhecimentos sobre o que é o stress e como poderá prevenir este em ambiente empresarial. Nesta grandeza serão utilizados os fatores <b>Aprendizagem e Avaliação</b> .
<b>Ergonómica</b>	Grandeza que irá permitir obter <i>feedback</i> sobre a interação do jogador com o jogo sério. Serão utilizados os fatores <b>Usabilidade, Jogabilidade e Sociocultural</b> .
<b>Técnica</b>	Grandeza que irá permitir verificar que as funcionalidades desenvolvidas encontram-se como pretendido para o jogador.

#### 7.4.1 Quantitative Evaluation Framework

O *Quantitative Evaluation Framework* (QEF) sendo um modelo criado com o objetivo de validar e garantir a qualidade de um software educativo (Paula & José, 2008). Neste projeto será utilizado o QEF de modo a avaliar a qualidade deste. No QEF criado serão utilizadas as dimensões referidas anteriormente, com os respetivos fatores Tabela 35.

Tabela 35. QEF Dimensões e Fatores

<b>Dimensão</b>	<b>Fator</b>
Pedagógica	Aprendizagem
	Avaliação
Ergonómica	Usabilidade
	Jogabilidade
	Sociocultural
Técnica	Suporte
	Funcionalidades

Na dimensão pedagógica, pretende-se avaliar se o projeto desenvolvido promove a aprendizagem, tal como uma avaliação.

Na próxima dimensão ergonómica pretende-se avaliar a interação entre o jogador e o jogo sério, ou seja, a usabilidade do jogo, tal como fatores externos ao jogador, por exemplo, se o jogo esta disponível em mais do que um idioma.

Por último, na dimensão técnica pretende-se avaliar as funcionalidades implementadas no jogo e o suporte que o jogo fornece ao jogador.

O QEF realizado contém 59 requisitos, nas quais cada um terá de ser autoavaliado pela autora do presente documento, no final do desenvolvimento, seguindo as métricas representadas no Anexo A. Algumas métricas irão depender das classificações obtidas no questionário SUS, que será descrito no subcapítulo seguinte.

Feito o descrito anteriormente, obteve-se o resultado de 95%, como se pode verificar na Figura 78. Sendo que dos 27 requisitos, pertencentes a componente técnica, não foram

desenvolvidos três, nas quais, a conexão entre o Unity e um aparelho de Biofeedback, no caso iria ser utilizado um *smartwatch*, o jogo utilizar o DDA, já explicado como seria aplicado no subcapítulo Dificuldade, e por fim, não foi aplicado um sistema de ajuda. A componente, apesar de faltarem requisitos, teve um total de 88,1% de implementações, já a componente ergonômica obteve 91,74%. A última componente, obteve um resultado de 100%.

q	D	α <sub>i</sub>	Dimensão	Q <sub>i</sub>	P <sub>ij</sub> (média do fator / P <sub>ij</sub> ) (R, G, B, V)	Fator	P <sub>ij</sub> (média de requisitos / P <sub>ij</sub> ) (R, G, B, V)	Requisitos	PC <sub>i</sub> (R de requisitos / R de Q <sub>i</sub> ) (R, G, B, V)
95%	0.14	88.1	Técnica	80.88	0.16	Suporte	8	O jogo é estável e executa sistematicamente sem falhas	100
							10	A instalação do jogo é fácil para o público-alvo	100
							8	Os dados do jogador são assegurados em caso de falha na aplicação	100
							8	O aplicativo garante conexão com aparelhos de Biofeedback (Smartwatch)	0
							10	O aplicativo garante compatibilidade com múltiplas Plataformas (Android, Windows)	100
							10	O aplicativo garante compatibilidade com múltiplos Dispositivos (Tablet, Smartphone, Computador)	50
							6	O jogo pode ser facilmente atualizado	100
							8	Há um ponto de entrada clara e simples para o jogo	100
							10	A apresentação dos conceitos é clara e oportuna	100
				8	Os efeitos das ações do jogador podem ser vistos claramente no ambiente do jogo	100			
				2	O jogo contém uma intro	100			
				10	O jogador terá acesso a um menu principal	100			
				10	O jogador terá acesso ao menu dos níveis	100			
				10	O jogador poderá iniciar o nível que pretende jogar	100			
				8	O jogador poderá aceder a coleção de cartas	100			
				8	O jogador poderá ver informação de cada carta	100			
				10	O jogador poderá realizar o questionário	100			
				10	O jogador poderá ver os resultados obtidos pela análise do questionário	100			
		6	O jogador poderá sair do jogo	100					
		8	O jogador poderá ver o manual	100					
		6	O jogador poderá ver informações sobre o jogo	100					
		8	O jogador poderá alterar o idioma	100					
		8	O jogador poderá modificar o volume do áudio do jogo	100					
		8	O jogador poderá utilizar o sistema de ajuda ("Hint")	0					
		8	O jogo utiliza DDA (Dynamic difficulty adjustment)	0					
		6	O jogo tem consistência entre as cores utilizadas	100					
		8	O jogo possui consistência no mecanismo de navegação	100					
		8	Os conteúdos apresentados contribuem para motivar o jogador	75					
		10	As instruções do jogo são claras, precisas e concisas	100					
		6	O jogador pode facilmente iniciar e sair do jogo	100					
		10	A interação com o jogo é intuitiva	100					
		8	O jogador pode usar o jogo sem ler o manual	100					
		6	Ícones do jogo é apropriado e consistente	100					
8	Um botão de ajuda é fornecido	0							
6	A navegação é consistente ao longo do jogo e é realizada facilmente	100							
8	A velocidade de comunicação entre o jogador e o jogo é adequada	100							
6	O conteúdo escrito/falado está livre de erros gramaticais e sintáticos.	100							
91.74	0.14	91.74	Ergonómica	93.33	0.44	Jogabilidade	8	O jogo é original	100
							10	O jogador controla as ações do jogo	100
							8	O uso de áudio melhora o jogo	100
							8	Gráficos e imagens melhoram o jogo	100
							8	O jogo é divertido de jogar	100
							10	O jogo possui um manual	100
							6	O jogo é desafiador	100
							8	O jogador tem um senso de controle sobre as suas ações e é capaz de usar táticas e estratégias	100
							8	O jogador sente que os resultados dos seus esforços/ações são justos	100
		6	O sistema de ajuda (destaque de objetos ou dicas) foi satisfatório para ajudar os jogadores	0					
		10	Os menus do jogo são facilmente acessíveis	100					
		6	O jogo está disponível em múltiplos idiomas	100					
		10	A linguagem do jogo é adequada ao público-alvo	100					
		10	O jogo não apresenta conteúdo ofensivo ou estereótipos em termos de gênero, raça, religião e diversidade	100					
		10	A Linguagem não inclui conteúdo que incentive à violência	100					
100	0.14	100	Pedagógica	100	0.50	Aprendizagem	10	O contexto de aprendizagem é adequado	100
							8	O objetivo de aprendizagem é esclarecido	100
							10	O jogo promove a autoaprendizagem	100
		8	O jogo permite o desenvolvimento cognitivo, para que o jogador possa utilizá-lo em outros contextos	100					
		8	A informação do jogo poderá ser utilizada no dia a dia do jogador	100					
		100	0.50	Avaliação	8	O jogo promove a autoavaliação	100		
6	O resultado da autoavaliação do será fornecida através de um gráfico	100							

Figura 78. Resultado QEF

## 7.5 System Usability Scale

Para avaliar alguns pontos do QEF foi realizado um questionário baseado no System Usability Scale (SUS). Este questionário foi lançado em 1986 por John Brook (Sauro, 2011), e é composto por 10 questões. A escala de avaliação é composta por 5 possíveis respostas, nas quais, 1 significa que o utilizador discorda fortemente, 2 que discorda, 3 é neutro, 4 concorda com a afirmação e por último 5, concorda fortemente com a afirmação (Figura 79).

Strongly Disagree 1	2	3	4	Strongly Agree 5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figura 79. Escala de avaliação SUS

Este questionário, como dito anteriormente, é composto por 10 perguntas nas quais, as afirmações de número ímpares são afirmativas, no entanto, a de números pares terão um lado negativo.

### 7.5.1 Questionário

Foi utilizado o *Microsoft forms*, de modo a conseguir criar e disponibilizar o questionário ao grupo de colaboradores que iriam testar o jogo. Na Tabela 36, encontram-se as afirmações selecionadas para o questionário de modo a avaliar a usabilidade do jogo, tal como outros pontos do jogo.

Tabela 36. Perguntas do Questionário.

Afirmação
A ideia do jogo é original
O jogo <b>não</b> é divertido de jogar, <b>nem</b> desafiante.
Fiquei motivado para continuar a jogar.
<b>Senti-me "preso"</b> num nível, durante o jogo.
O jogo não é intrusivo e é fácil de entender.
O som e os gráficos do jogo <b>não</b> me ajudaram a mergulhar no ambiente.
Consegui iniciar o jogo, sem a necessidade de ver o manual desde o menu inicial até sair do jogo.
<b>Não</b> consegui ter controlo sobre as minhas ações e usa estratégias, sendo que no final os esforços <b>não</b> foram justos.
Achei que o conteúdo do jogo é apropriado, não contendo quaisquer conteúdos que incentive a violência ou ofensivo em termos de gênero, raça, religião e diversidade cultural.
A informação disponibilizada no jogo, <b>não</b> permitiu autoaprendizagem, <b>nem</b> que os conhecimentos sejam utilizados para outros contextos ou até mesmo no meu dia a dia.

### 7.5.2 Resultados de cada afirmação

Serão apresentados, neste subcapítulo, os resultados e análise de cada resposta e o processo necessário para a obtenção dos dados.

Como referido anteriormente, o questionário foi respondido por 10 colaboradores, e obteve-se o resultado da Tabela 37. Esta demonstra o número de vezes que uma opção foi selecionada, para cada questão.

Tabela 37. Número de vezes ocorrido uma avaliação.

Pergunta/pontuação	1	2	3	4	5
1	0	0	1	7	2
2	3	7	0	0	0
3	0	0	3	5	2
4	0	1	7	2	0
5	0	0	0	2	8
6	4	5	1	0	0
7	7	3	0	0	0
8	7	3	0	0	0
9	0	0	0	0	10
10	6	3	1	0	0

Considerando os valores quatro e cinco, valores positivos, o três, valor neutro, e o dois e um, valores negativos, foi realizado um gráfico para cada pergunta de modo a concluir a percentagem de valores, ou seja, verificar se foi uma avaliação positiva, negativa ou neutra.

No primeiro resultado, obtiveram-se 90% de valores positivos, tendo uma avaliação geral positiva (Gráfico 1), sendo assim, pode-se considerar que a ideia do jogo é original.

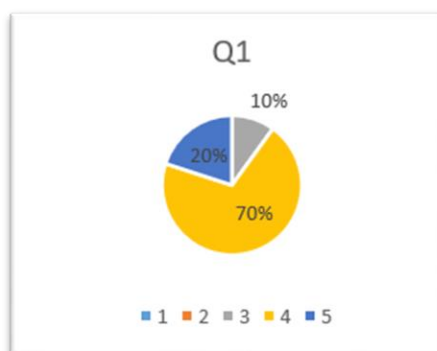


Gráfico 1. Resultado da afirmação “A ideia do jogo é original”.

Na segunda afirmação, obtiveram-se 100% de valores positivos (Gráfico 2), sendo assim, o grupo de colaboradores considera que o jogo é desafiante e divertido de jogar.

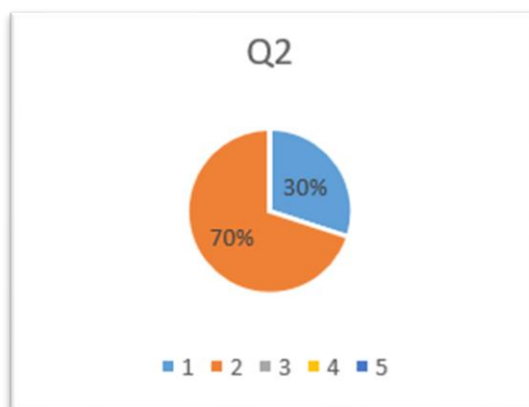


Gráfico 2. Resultado da afirmação “O jogo não é divertido de jogar, nem desafiante”.

No caso da terceira afirmação, Gráfico 3, obteve-se uma avaliação de 70%, uma avaliação positiva, sendo assim, os jogadores sentiram-se motivados para continuar a jogar.

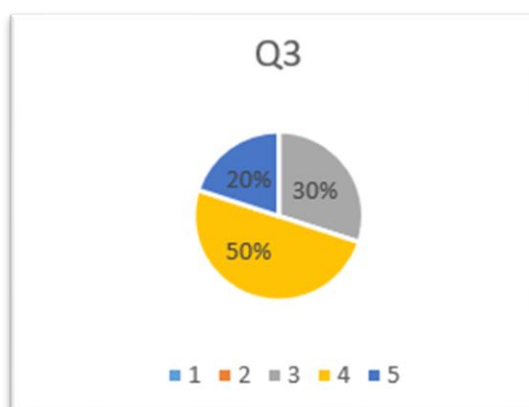


Gráfico 3. Resultado da afirmação “Fiquei motivado para continuar a jogar”.

Na quarta afirmação, Gráfico 4, obteve-se um total de 70% em avaliação neutra, os jogadores, tiveram uma resposta neutra em relação a ficarem “presos” num determinado nível. Este será um tema que deverá ser melhorado futuramente.

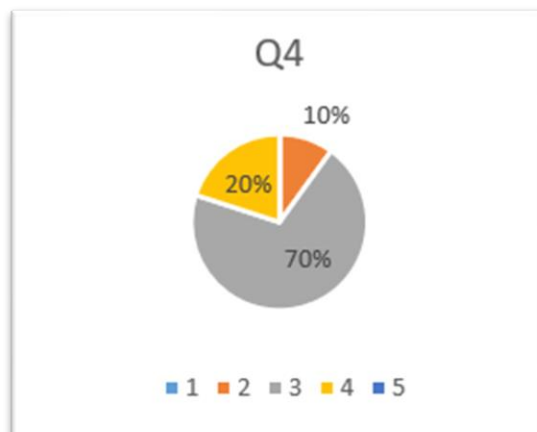


Gráfico 4. Resultado da afirmação "Senti-me "preso" num nível, durante o jogo".

Na afirmação seguinte obteve-se uma avaliação de 100% Positiva (Gráfico 5), ou seja, os jogadores consideram que o jogo não é intrusivo e este foi fácil de entender.

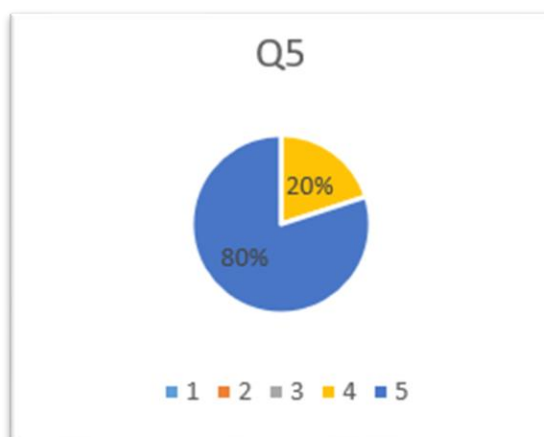


Gráfico 5. Resultado da afirmação "O jogo não é intrusivo e é fácil de entender".

Na sexta afirmação, como se pode verificar no Gráfico 6, o feedback foi positivo, sendo assim, pode-se concluir o jogador considera que o som e os gráficos do jogo são agradáveis.

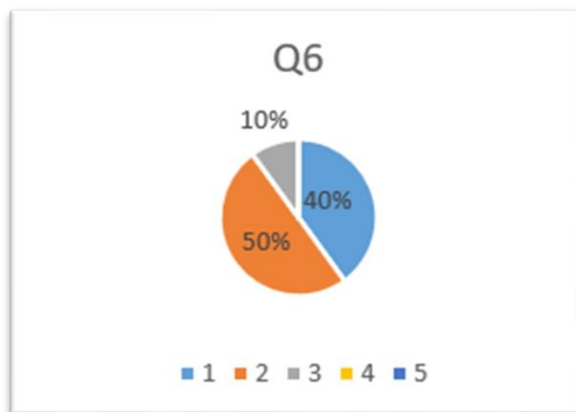


Gráfico 6. Resultado da afirmação relacionada com o som e os gráficos do jogo

A sétima afirmação, obtiveram-se 80% de feedback positivo, como se pode verificar no Gráfico 7, pode-se concluir, com este, que os jogadores iniciaram facilmente o jogo sem a necessidade de ler o manual.

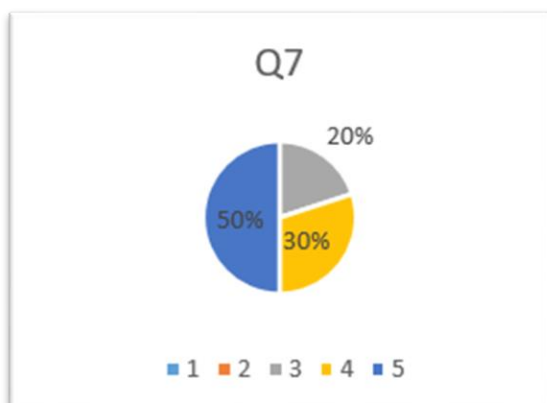


Gráfico 7. Resultado da afirmação relacionada com o início do jogo

Na próxima afirmação, como se pode verificar no Gráfico 8, os jogadores consideram que tiveram controlo sobre as ações destes, avaliando positivamente o jogo.

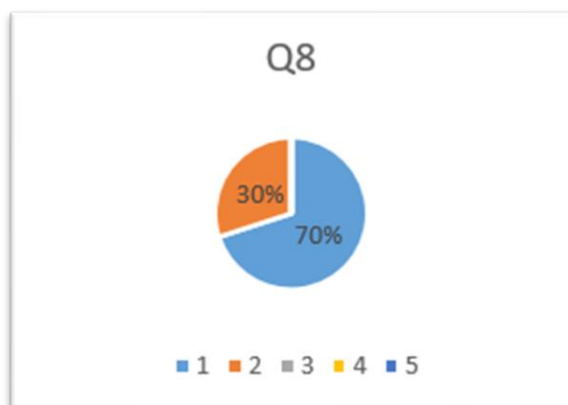


Gráfico 8. Resultado da afirmação sobre o controlo do jogo.

Na penúltima afirmação, obteve-se feedback positivo a 100% (Gráfico 9), sendo assim, pode-se concluir que o conteúdo do jogo é apropriado.

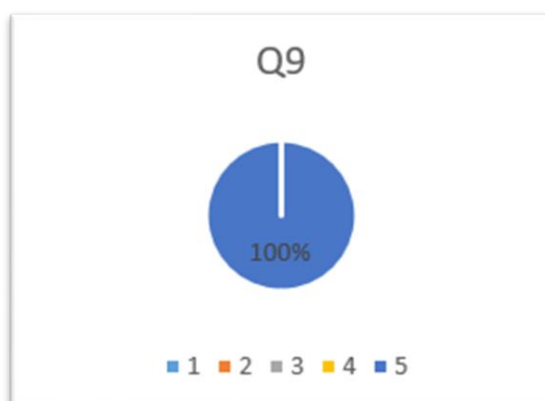


Gráfico 9. Resultado da afirmação relacionada com o conteúdo.

Na última afirmação, obteve-se feedback positivo, com 70% de respostas positivas (Gráfico 10). Em suma, os colaboradores consideram que a informação disponibilizada no jogo permitiu a autoaprendizagem e que os conhecimentos adquiridos possam ser utilizados para outros contextos ou até mesmo no meu dia a dia.

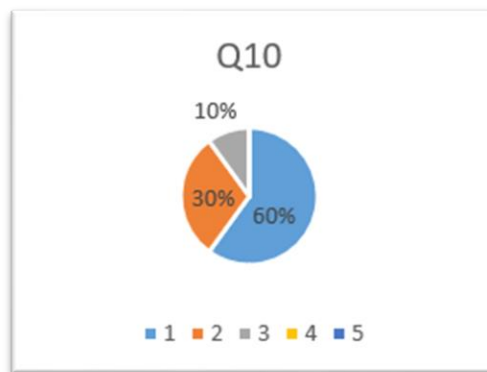


Gráfico 10. Resultado a última afirmação.

### 7.5.3 Cálculo final

Neste subcapítulo será descrito como se obteve o resultado do questionário SUS. Tendo em consideração o método de cálculo seguinte:

- Afirmações Ímpares:  $(x - 1)$ , com  $x$  sendo o valor da avaliação
- Afirmações Pares:  $(5 - x)$

Primeiro passo realizado, foi calcular a primeira fase do questionário SUS (Figura 80). Este, é executado sobre a linha de respostas de um colaborador, utilizando o método de cálculo referido anteriormente.

$$1 = (B2-1)+(5-C2)+(D2-1)+(5-E2)+(F2-1)+(5-G2)+(H2-1)+(5-I2)+(J2-1)+(5-K2)$$

Figura 80. Cálculo executado por linha

Segundo passo seria, passar os valores obtidos para a escala de SUS, ou seja, numa escala de 0 a 100, para tal, cada resultado foi multiplicado por 2,5 (Figura 81). Por último, foi executada a média de valores, obtendo a média de 82.

Nº da Linha	Cálculo por linha	Resultado na escala do SUS
1	33	82.5
2	35	87.5
3	32	80
4	34	85
5	34	85
6	31	77.5
7	32	80
8	36	90
9	32	80
10	29	72.5
Média Final		82

Figura 81. Resultado dos Cálculos SUS.

De acordo com a Figura 82, que representa a escala de avaliação do questionário, o projeto desenvolvido encontra-se na nota A, ou seja, uma avaliação **excelente**.

SUS Score	Grade	Adjective Rating
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay
51 – 68	D	Poor
< 51	F	Awful

Figura 82. Escala de avaliação SUS.

## 7.6 Apreciação dos Resultados Obtidos

Após a análise anteriormente, é necessário aferir qual das hipóteses referidas na secção 7.1, o protótipo desenvolvido se enquadra. Com os resultados obtidos através do questionário System Usability Scale, pode-se confirmar que a Hipótese Alternativa (H1) é a que mais se adequa. Existem, no entanto, possíveis melhorias a serem implementadas, estas serão descritas na Conclusão da dissertação.

## 8 Conclusão

Neste capítulo será realizada uma validação de modo a verificar se os objetivos, descritos na secção Objetivos foram executados. Será, também, executado um levantamento de possíveis melhorias a serem implementadas futuramente. Por último, de modo a finalizar o documento será executada uma apreciação final sobre toda a etapa desde a escrita da dissertação até ao final da realização do protótipo.

### 8.1 Objetivos

Antes de se referir possíveis alterações e até mesmo ser feita uma apreciação final, é necessário validar se os objetivos referidos na secção 1.2 foram cumpridos. Segue-se a lista já identificada.

- Identificar e analisar o problema;
- Identificar alternativas existentes no mercado;
- Desenvolver um jogo sério que auxilie no combate e na prevenção do stress em ambientes profissionais;
- Experimentação e avaliação do projeto desenvolvido.

Para o primeiro e o segundo objetivo serem cumpridos, foram executados os seguintes passos. Primeiro, executada uma pesquisa profunda sobre o problema que se pretende solucionar e aliviar, tal como, um estudo das alternativas que existem atualmente no mercado. Esta informação foi apresentada na secção Estado da Arte.

No caso do terceiro e quarto objetivo, foi necessário executar um desenho de uma possível solução, este foi apresentado na secção 5. Após esta fase ter sido terminada, foi desenvolvido um protótipo e executou-se a sua avaliação utilizando o QEF, já descrito na secção número 7. Através desta avaliação obteve-se, no final, 88% na componente técnica, e 95% em relação a todos os componentes.

### 8.1.1 Sucesso do protótipo

Como já referido anteriormente, sendo 95% o resultado da avaliação utilizando o QEF e o resultado de **excelência**, obtido utilizando o questionário SUS, apresentado na secção número 7.5, pode-se concluir que o protótipo teve um resultado deveras positivo.

## 8.2 Trabalho Futuro

No questionário disponibilizado aos colaboradores, apresentado na secção System Usability Scale, permitia darem um feedback de modo descritivo. Após obtidas as respostas, a autora realizou um resumo, acrescentando informação que considerou crucial. Segue-se a lista de possíveis melhorias e implementações futuras:

- **Melhoria na qualidade de resolução da introdução:**

Um dos feedbacks mais comum era relacionado com a qualidade da introdução do jogo. Sendo um *software* grátis, a qualidade de resolução implementada não foi a desejada. Para implementar esta melhoria, será necessário utilizar um determinado orçamento de modo a elevar a qualidade.

- **Cartas e Coleção:**

Atualmente o jogo tem tradução em todos os menus excluindo, no entanto, as cartas que o jogador possui. Esta seria uma melhoria a possível implementar futuramente.

Para além do descrito anteriormente, quando um jogador obtém uma carta não existe nenhuma animação. Outra melhoria seria, portanto, a implementação de uma animação assim que o jogador vence um nível e desbloqueia uma nova carta. Tal como aumentar a qualidade de imagens utilizadas nestas.

- **Informação das Cartas**

Em relação à informação disponibilizada nas cartas, poderia ser executada uma parceria com profissionais da área de saúde de modo a abordar mais sobre o problema e possíveis métodos de prevenção deste. Esta colaboração permitiria também permitindo garantir que toda a informação apresentada seria cientificamente fiável e o mais clara possível.

- **Questionário e Resultados:**

Em relação ao questionário disponibilizado no jogo, o resultado de cada questão, em vez de um *dropdown* poderia ser substituído por um conjunto de botões, por exemplo, por *smiles*. Outra possível implementação, seria o aparecimento do questionário caso um jogador tivesse decidido desistir de um dos níveis, durante um período, deste modo a verificar o nível de stress naquele momento.

A informação obtida do questionário, referido anteriormente, é disponibilizada através de um gráfico que atualmente não faz referência ao dia em que o jogador

executa o questionário, mas sim ao número do dia. A referência ao dia em que o jogador executa o questionário poderia ser uma das melhorias a efetuar. Para além disso, disponibilizar antigos gráficos, por exemplo, no mês anterior, e permitir executar uma limpeza de dados após um período.

- **Biofeedback:**

Em relação ao biofeedback, a autora pretende desenvolver uma solução para conseguir a compatibilidade do jogo com um *wearable*. Deste modo a obtenção dos resultados sobre o nível de stress seriam através deste método de *feedback*, o questionário seria outra opção, para quem não possui um *wearable*.

- **Botão de ajuda nos níveis:**

Implementação de um botão de ajuda.

- **Pop-ups:**

Uma melhoria a ser implementada futuramente seria a diminuição da quantidade de *Pop-ups* que aparecem atualmente.

- **Base de dados:**

Como referido na secção Protótipo Desenvolvido, não foi utilizado a ligação a uma base de dados, sendo esta, uma melhoria a ser implementada no protótipo, de modo a evitar a utilização de ficheiros.

### 8.3 Apreciação final

Ao longo do desenvolvimento desta dissertação, a autora foi se apercebendo da importância dos jogos sérios na saúde e de como estes podem proporcionar uma mudança da vida dos jogadores. Como tal continua a haver motivação, por parte da autora, para a continuidade do projeto, de modo a conseguir, um dia em breve, fornecer o jogo para ser aplicado nos meios cooperativos.

Para além disso, este projeto serviu para a autora desenvolver conhecimentos em diversas e distintas áreas. Adquirindo conhecimentos, como, por exemplo, na metodologia de pesquisa, na capacidade de analisar um problema e posteriormente desenhar e construir uma solução para combater um problema, e executar uma avaliação de acordo com os diversos métodos de avaliação.



# Referências

- (s.d.). Obtido em 3 de janeiro de 2022, de Spire Health:  
<https://www.spirehealth.com/technology>
- (s.d.). Obtido em 3 de janeiro de 2022, de Muse: <https://choosemuse.com>
- A.Koen, P., Ajamian, G., & S. B. (s.d.). Fuzzy Front End:. Obtido de  
[https://moodle.isep.ipp.pt/pluginfile.php/179206/mod\\_folder/content/0/FuzzyFrontEnd\\_Old.pdf?forcedownload=1](https://moodle.isep.ipp.pt/pluginfile.php/179206/mod_folder/content/0/FuzzyFrontEnd_Old.pdf?forcedownload=1)
- altexsoft*. (s.d.). Obtido de Non-functional Requirements: Examples, Types, How to Approach:  
<https://www.altexsoft.com/blog/non-functional-requirements/>
- B. S., & P. S. (19 de fevereiro de 2008). *Serious Games Taxonomy*. Obtido em 30 de dezembro de 2020, de The Digital Entertainment Alliance:  
<https://thedigitalentertainmentalliance.files.wordpress.com/2011/08/serious-games-taxonomy.pdf>
- BROWN, K. (19 de novembro de 2019). *What Is GitHub, and What Is It Used For?* Obtido em 30 de dezembro de 2021, de How-to Geek: <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>
- C. O., & M. B. (21 de maio de 2008). Considerações sobre a obtenção de vetores de prioridades no AHP. Obtido em 06 de fevereiro de 2021, de  
[https://cleberalmeida.webs.com/T51AVANCES\\_TEORICOS\\_BELDERRAIN.pdf](https://cleberalmeida.webs.com/T51AVANCES_TEORICOS_BELDERRAIN.pdf)
- Canvas de Proposta de Valor: passo a passo para criar o seu*. (30 de junho de 2020). Obtido em 28 de dezembro de 2021, de empresa jr: <https://empresajrespm.com.br/canvas-de-proposta-de-valor-passo-a-passo-para-criar-o-seu/>
- Casarotto, C. (20 de dezembro de 2019). *Aprenda o que é análise SWOT*. Obtido em 30 de dezembro de 2021, de Rock Content Blog: <https://rockcontent.com/br/blog/como-fazer-uma-analise-swot/>
- Cherry, K. (11 de julho de 2021). *What Is Biofeedback?* Obtido em 30 de dezembro de 2021, de Very Well Mind: <https://www.verywellmind.com/what-is-biofeedback-2794875>
- Cohen, S. (s.d.). PERCEIVED STRESS SCALE. Obtido em 30 de janeiro de 2022, de  
<https://www.sprc.org/system/files/private/event-training/Penn%20College%20-%20Perceived%20Stress%20Scale.pdf>
- Comfy Sacks. (2020). *Most Stressful Video Games*. Obtido em 30 de janeiro de 2022, de Comfy Sacks: <https://www.comfysacks.com/most-stressful-video-games>
- Conceito de jogo*. (2020). Obtido em 30 de janeiro de 2021, de Conceito de:  
<https://conceito.de/jogo>

- Desdobramento da Função Qualidade (QFD)*. (s.d.). Obtido em 3 de fevereiro de 2022, de Info escola: [https://www.infoescola.com/administracao\\_/desdobramento-da-funcao-qualidade-qfd/](https://www.infoescola.com/administracao_/desdobramento-da-funcao-qualidade-qfd/)
- DESK, F. N. (12 de abril de 2019). *FRIEZE*. Obtido de Blue is the Most Relaxing Colour, Scientists Say: <https://www.frieze.com/article/blue-most-relaxing-colour-scientists-say>
- Eatough, E. (13 de setembro de 2021). *How stress trackers can help you stay calm*. Obtido em 21 de dezembro de 2021, de BetterUp: <https://www.betterup.com/blog/stress-tracker>
- EBUN-AMU, C. (16 de março de 2021). *What Is Dynamic Difficulty Adjustment, and Why Does EA Own the Patent?* Obtido em 30 de janeiro de 2022, de Make Use Of: <https://www.makeuseof.com/what-is-dynamic-difficulty-adjustment/>
- Eduardo, D. (8 de abril de 2015). *Porque Utilizar o Unity ?* Obtido em 10 de fevereiro de 2021, de Eng: <https://www.eng.com.br/artigo.cfm?id=17>
- Fernandes, M. (9 de fevereiro de 2016). *PlayerPrefs Unity3D*. Obtido em 27 de fevereiro de 2021, de Matheus Fernandes: <https://doctorzeroth.wordpress.com/2016/02/09/playerprefs-unity3d/>
- Fidelis, R. (31 de dezembro de 2018). *As 10 melhores Game Engine para criar seu próprio jogo do zero*. Obtido em 20 de janeiro de 2021, de Criar Um Jogo: <https://criarumjogo.com/game-engine/>
- Games, H. L. (8 de janeiro de 2016). *Unity Asset Store*. Obtido de Sliding Tile Puzzle Game: <https://assetstore.unity.com/packages/templates/packs/sliding-tile-puzzle-game-41798>
- Healing, P. (s.d.). *Ondas Cerebrais | Entenda o que é a Onda Theta!* Obtido em 1 de fevereiro de 2020, de PortalHealing: <https://portalhealing.com.br/2020/09/25/ondas-cerebrais-entenda-o-que-e-a-onda-theta/>
- Hepburn, D. D. (30 de abril de 2019). *The Difference Between Positive and Negative Stress*. Obtido em 30 de julho de 2022, de medium: <https://medium.com/@Dr.DavidHepburn/the-difference-between-positive-and-negative-stress-de81d9596643>
- House Of Quality. (s.d.). Obtido de [https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjs3YvX14z2AhUw\\_rslHaKABJgQFnoECAgQAQ&url=https%3A%2F%2Fwww.ieee.li%2Ftmc%2Ftraditionalhoq.xlt&usg=AOvVaw1IUJSnZPz2oS9HdAX0W1VJ](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjs3YvX14z2AhUw_rslHaKABJgQFnoECAgQAQ&url=https%3A%2F%2Fwww.ieee.li%2Ftmc%2Ftraditionalhoq.xlt&usg=AOvVaw1IUJSnZPz2oS9HdAX0W1VJ)
- How To Measure Stress*. (s.d.). Obtido em 30 de dezembro de 2021, de Stress Resilient Mind: <https://www.stressresilientmind.co.uk/articles/how-to-measure-stress>

- IEEE. (31 de março de 2016). *IEEE Xplore*. Obtido em 30 de janeiro de 2021, de Ubiquitous Biofeedback Serious Game for Stress Management: <https://ieeexplore.ieee.org/document/7445149>
- J. H., E. S., & B. R. (junho de 2015). *Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International*. Obtido em 6 de fevereiro de 2021, de ResearchGate Logo: [https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.researchgate.net%2Ffigure%2FAHP-scale-Pairwise-comparison-scale\\_tbl1\\_277726029&psig=AOvVaw15fSxsLc1wNhK0mkEJoPjd&ust=1613579033585000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCNj19L7o7u4CFQAAAAAdAAAAABAD](https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.researchgate.net%2Ffigure%2FAHP-scale-Pairwise-comparison-scale_tbl1_277726029&psig=AOvVaw15fSxsLc1wNhK0mkEJoPjd&ust=1613579033585000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCNj19L7o7u4CFQAAAAAdAAAAABAD)
- Junior, D. d. (outubro de 2002). A competição como fonte de estresse no esporte. *Competition as a source of stress in sport*, p. 8. Obtido em 16 de Junho de 2022
- Leaf Urban*. (s.d.). Obtido em 3 de janeiro de 2022, de Bella Beat: <https://bellabeat.com/leaf-urban/>
- MongoDB Atlas*. (s.d.). Obtido em 27 de fevereiro de 2021, de MongoDB: <https://docs.atlas.mongodb.com>
- P. E., & J. B. (janeiro de 2008). *Quantitative Evaluation Framework (QEF)*. Obtido em 10 de fevereiro de 2021, de ResearchGate: [https://www.researchgate.net/publication/257579051\\_Quantitative\\_Evaluation\\_Framework\\_QEF](https://www.researchgate.net/publication/257579051_Quantitative_Evaluation_Framework_QEF)
- Pereira, D. (5 de março de 2019). *Canvas da Proposta de Valor*. Obtido em 28 de dezembro de 2021, de O Analista de Modelos de Negócios: <https://analistamodelosdenegocios.com.br/canvas-da-proposta-de-valor/>
- Pereira, R. E. (julho de 2017). *Stress Ocupacional: Perspetiva Antropológica na Restauração*. Coimbra, Portugal. Obtido em 30 de Janeiro de 2021, de <https://estudogeral.sib.uc.pt/bitstream/10316/83124/1/Dissertação%20Rúben%20Pereira%20Final.pdf>
- Portugal, T. (s.d.). Obtido em 10 de fevereiro de 2021, de Talent Portugal: <https://talentportugal.com/beneficios/emprego-sala-de-jogos/>
- Proposta de valor*. (s.d.). Obtido em 31 de janeiro de 2021, de Dicionário Financeiro: <https://www.dicionariofinanceiro.com/proposta-de-valor/>
- Rodrigues, L. F. (12 de outubro de 2020). *Seis videojogos para combater o stress*. Obtido em 30 de janeiro de 2021, de Time Out: <https://www.timeout.pt/lisboa/pt/coisas-para-fazer/jogos-anti-stress>

- Santos, B. L. (s.d.). *STRESS NO TRABALHO: CAUSAS E CONSEQUÊNCIAS*. Obtido em 3 de janeiro de 2022, de Alento: <https://www.alento.pt/noticias/Stress-no-trabalho:-Causas-e-Consequencias>
- Sauro, J. (2 de fevereiro de 2011). *MeasuringU*. Obtido em 20 de janeiro de 2021, de MEASURING USABILITY WITH THE SYSTEM USABILITY SCALE (SUS): <https://measuringu.com/sus/#>
- Serious games for police officers under stress*. (15 de março de 2017). Obtido em 30 de janeiro de 2021, de International Alumni: <https://www.ru.nl/alumni/english/@1086113/games-police-officers-stress/>
- Silva, I. (29 de agosto de 2017). *Reduzir o stress no trabalho com jogos? Explicamos tudo*. Obtido em 10 de fevereiro de 2021, de Economista: <https://www.economista.pt/reduzir-o-stress-no-trabalho-com-jogos/>
- Soares, J. (2018). *RELOAD Menos Stress. Melhor Performance*. Porto: Porto Editora.
- Staff, F. (5 de julho de 2017). *Teletrabalho é mais stressante, diz inquérito*. Obtido em 6 de dezembro de 2021, de Portugal Forbes: <https://www.forbespt.com/teletrabalho-e-mais-stressante-diz-inquerito/>
- Technology, P. (s.d.). *What Is The Analytic Hierarchy Process (AHP)?* Obtido em 6 de fevereiro de 2021, de Passage Technology: <https://www.passagetechnology.com/what-is-the-analytic-hierarchy-process>
- Teixeira, D. (10 de fevereiro de 2018). *Como escrever requisitos de software de forma simples e garantir o mínimo de erros no sistema/app??* Obtido em 27 de fevereiro de 2021, de Medium: <https://medium.com/lfdev-blog/como-escrever-requisitos-de-software-de-forma-simples-e-garantir-o-mínimo-de-erros-no-sistema-app-74df2ee241cc>
- TEST YOUR STRESS*. (s.d.). Obtido em 20 de janeiro de 2022, de Be Mindful: <https://www.bemindfulonline.com/test-your-stress>
- Trabalho, A. E. (27 de junho de 2014). *Cálculo do custo do stresse e dos riscos psicossociais relacionados com o trabalho*. Obtido em 30 de janeiro de 2020, de Publicações: <https://osha.europa.eu/pt/publications/calculating-cost-work-related-stress-and-psychosocial-risks/view>
- Trabalho, A. E. (s.d.). *Riscos psicossociais e stresse no trabalho*. Obtido em 30 de janeiro de 2021, de Agência Europeia para a Segurança e Saúde no Trabalho: <https://osha.europa.eu/pt/themes/psychosocial-risks-and-stress>
- Trello*. (s.d.). Obtido em 30 de dezembro de 2021, de Trello: <https://trello.com>
- Versus Headset*. (s.d.). Obtido em 3 de janeiro de 2022, de Versus: <https://getversus.com/headset>

Zohaib, M. (14 de novembro de 2018). *Hindawi*. Obtido em 15 de julho de 2022, de Dynamic Difficulty Adjustment (DDA) in Computer Games: A Review: <https://www.hindawi.com/journals/ahci/2018/5681652/>



# Anexo A – Métricas do QEF

## Componente Técnica

Fator:						
Suporte						
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
		0	25	50	75	100
O jogo é estável e executa sistematicamente sem falhas	O desenvolvedor irá testar o jogo, de modo a validar se executa sem falhas.	Possui falhas sistematicamente	-	Possui, no máximo 2 falhas.	-	Não possui falhas
A instalação do jogo é fácil para o público-alvo	Se o jogo é fácil e rápido de instalar, para utilizadores não experientes.	Não	-	-	-	Sim
Os dados do jogador são assegurados em caso de falha na aplicação	Os dados serão validados durante os testes, avigorando que estas são os que o jogador guardou.	Não são assegurados	-	-	-	São assegurados
O aplicativo garante conexão com aparelhos de Biofeedback (Smartwatch)	O jogo deverá ter compatibilidade com um aparelho de biofeedback.	Não implementado	-	-	-	Implementado
O aplicativo garante compatibilidade com múltiplas Plataformas (Android, Windows)	O desenvolvedor irá testar o jogo em diferentes plataformas	Não implementado	-	Compatibilidade com 1 plataforma	-	Implementado
O aplicativo garante compatibilidade com múltiplos Dispositivos (Tablet, Smartphone, Computador)	O desenvolvedor irá testar o jogo em diferentes dispositivos.	Não implementado	-	Compatibilidade com 2 Dispositivos	-	Implementado
O jogo pode ser facilmente atualizado	O jogo deverá ser facilmente atualizado pelo jogador.	Não	-	-	-	Sim
Há um ponto de entrada clara e simples para o jogo	O jogo deverá ter apenas uma entrada, que seja simples e clara.	Não	-	-	-	Sim

Fator:						
Funcionalidades						
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
		0	25	50	75	100
A apresentação dos conceitos é clara e oportuna	Os conceitos são apresentados de forma clara, simples e oportuna.	Não	-	-	-	Sim
Os efeitos das ações do jogador podem ser vistos claramente no ambiente do jogo	O jogador pode ver os efeitos das suas ações no jogo (por exemplo, o aumento do volume).	Não implementado	-	-	-	Implementado
O jogo contém uma intro	O jogo possui uma introdução, com logo deste.	Não implementado	-	-	-	Implementado
O jogador terá acesso a um menu principal	O jogo possui um menu principal, após a entrada do jogador neste.	Não implementado	-	-	-	Implementado
O jogador terá acesso ao menu dos níveis	O jogador pode ver o menu dos níveis	Não implementado	-	-	-	Implementado
O jogador poderá iniciar o nível que pretende jogar	O jogador pode iniciar o nível que pretende jogar	Não implementado	-	-	-	Implementado
O jogador poderá aceder a coleção de cartas	O jogador pode aceder a coleção de cartas	Não implementado	-	-	-	Implementado
O jogador poderá ver informação de cada carta	O jogador pode ver informação de cada carta	Não implementado	-	-	-	Implementado
O jogador poderá realizar o questionário	O jogador pode realizar o questionário	Não implementado	-	-	-	Implementado
O jogador poderá ver os resultados obtidos pela análise do questionário	O jogador pode ver os resultados obtidos pela análise do questionário	Não implementado	-	-	-	Implementado
O jogador poderá sair do jogo	O jogador pode sair do jogo	Não implementado	-	-	-	Implementado
O jogador poderá ver o manual	O jogador pode ver o manual	Não implementado	-	-	-	Implementado
O jogador poderá ver informações sobre o jogo	O jogador pode ver informações sobre o jogo	Não implementado	-	-	-	Implementado
O jogador poderá alterar o idioma	O jogador pode alterar o idioma	Não implementado	-	-	-	Implementado
O jogador poderá modificar o volume do áudio do jogo	O jogador pode modificar o volume do áudio do jogo	Não implementado	-	-	-	Implementado
O jogador poderá utilizar o sistema de ajuda ("Hint")	O jogador pode utilizar o sistema de ajuda ("Hint")	Não implementado	-	-	-	Implementado
O jogo utiliza DDA (Dynamic difficulty adjustment)	O jogo utiliza DDA adequado para o jogo.	Não implementado	-	-	-	Implementado
O jogo tem consistência entre as cores utilizadas	O jogo tem consistência entre as cores utilizadas	Não tem consistência	-	-	-	Tem consistência
O jogo possui consistência no mecanismo de navegação	O jogo possui consistência no mecanismo de navegação, ou seja, o mecanismo de navegação é sempre o mesmo.	Não implementado	-	-	-	Implementado

## Componente Ergonómica

Fator:						
Usabilidade						
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
		0	25	50	75	100
Os conteúdos apresentados contribuem para motivar o jogador	O jogador considera que o conteúdo apresentado no jogo, o motiva para continuar. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
As instruções do jogo são claras, precisas e concisas	O jogador considera que as instruções do jogo são claras, precisas e concisas. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogador pode facilmente iniciar e sair do jogo	O jogador considera que é fácil iniciar e sair do jogo. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
A interação com o jogo é intuitiva	O jogador considera que é a interação com o jogo é intuitiva. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogador pode usar o jogo sem ler o manual	O jogador conseguiu iniciar o jogo, sem ler o manual. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
Simbolos do jogo é apropriado e consistente	Todos os símbolos, se repetidos são consistentes e apropriados.	Não implementado	-	-	-	Implementado
Um botão de ajuda é fornecido	O jogo contém um botão de ajuda, neste caso será, um botão de dica.	Não implementado	-	Implementado, mas sem função	-	Implementado
A navegação é consistente ao longo do jogo e é realizada facilmente	O jogador considera que a navegação é consistente e fácil de realizar. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
A velocidade de comunicação entre o jogador e o jogo é adequada	Cronometrar o tempo de resposta que o jogo demora, após o jogador executar uma ação.	Menor que 1s a responder	-	-	-	Menor que 1s a responder
O conteúdo escrito/falado está livre de erros gramaticais e sintáticos.	Não foram encontrados erros gramaticais, nem sintáticos pelo desenvolvedor, no final dos testes.	Contem erros gramaticais e sintáticos	-	-	-	Não contem erros gramaticais

Fator:		Wfk - Cumprimento (%)				
Jogabilidade		0	25	50	75	100
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
O jogo é original	O jogador considera o jogo original. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogador controla as ações do jogo	O jogador terá de controlar todas as ações do jogo.	Não controla as ações do jogo	-	-	-	80 – 100% de respostas positivas
O uso de áudio melhora o jogo	O jogador considera que o áudio melhor a experiência do jogo (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
Gráficos e imagens melhoram o jogo	O jogador considera que os gráficos e imagens melhoram a experiência do jogo. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogo é divertido de jogar	O jogador considera que é divertido o jogo. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogo possui um manual	O jogo possui um manual interno.	Não implementado	-	-	-	Implementado
O jogo é desafiador	O jogador considera que o jogo é desafiador (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogador tem um senso de controle sobre as suas ações e é capaz de usar táticas e estratégias	O jogador considera que tem controle pelas suas ações. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogador sente que os resultados dos seus esforços/ações são justos.	O jogador considera que os seus resultados/ações são justos. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O sistema de ajuda (destaque de objetos ou dicas) foi satisfatório para ajudar os jogadores	O sistema de ajuda foi o suficiente e satisfatório para ajudar o jogador. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
Os menus do jogo são facilmente acessíveis	O jogador considera que o menu do jogo é facilmente acessível. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas

Fator:		Wfk - Cumprimento (%)				
Sociocultural		0	25	50	75	100
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
O jogo está disponível em múltiplos idiomas	O jogo precisa de ter pelo menos 5 idiomas, a escolha do projeto	Só contem um idioma	Contem 2 idiomas	Contem 3 idiomas	Contem 4 idiomas	Contem 5 idiomas
A linguagem do jogo é adequada ao público-alvo	A linguagem tem de ser adequada onde o jogo será implementado.	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogo não apresenta conteúdo ofensivo ou estereótipos em termos de género, raça, religião e diversidade cultural	O jogo não faz qualquer referência a conteúdos que possam ser ofensivos ou estereótipos, em termo de género, raça, religião e diversidade cultural.	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
A linguagem não inclui conteúdo que incentive à violência	O jogo não faz qualquer referência nem incentiva a violência.	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas

## Componente Pedagógica

Fator:		Wfk - Cumprimento (%)				
Aprendizagem		0	25	50	75	100
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
O contexto de aprendizagem é adequado	Os temas abordados sobre o stress são adequados ao jogo.	Não	-	-	-	Sim
O objetivo de aprendizagem é esclarecido	O problema é descrito nas informações do jogo.	Não implementado	-	-	-	Implementado
O jogo promove a autoaprendizagem	O jogo fornece informações sobre o problema a resolver, tal como maneiras para evitar, ao longo do jogo. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
O jogo permite o desenvolvimento cognitivo, para que o jogador possa utilizá-lo em outros contextos	O jogo fornece informações e curiosidades sobre como prevenir o problema, e do que se trata este, onde o jogador poderá aplicar essa informação noutro contexto. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas
A informação do jogo poderá ser utilizada no dia a dia do jogador	O jogo fornece informação que poderá ser utilizada no dia a dia do jogador. (Questionário)	0 – 20% de respostas positivas	20 – 40% de respostas positivas	40 – 60% de respostas positivas	60 – 80% de respostas positivas	80 – 100% de respostas positivas

Fator:		Wfk - Cumprimento (%)				
Avaliação		0	25	50	75	100
Requisito	Métrica de Avaliação	Wfk - Cumprimento (%)				
O jogo promove a autoavaliação	O jogo fornece um mecanismo de autoavaliação, de modo a verificar se o objetivo do jogo esta a ser cumprido.	Não implementado	-	-	-	Implementado
O resultado da autoavaliação do será fornecida através de um gráfico	O resultado das autoavaliações será fornecido, como feedback ao jogador, através de um gráfico.	Não implementado	-	Implementado um feedback, mas sem a utilização de gráficos	-	Implementado

# Anexo B – Resultado do Questionário

! Sou uma pessoa stressada (0 ponto)

[Mais Detalhes](#)

[Informações](#)

● Concordo	4
● Discordo	6



A ideia do jogo é original. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

4.70  
Número Médio

O jogo **não** é divertido, **nem** desafiante. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

1.70  
Número Médio

Fiquei motivado para continuar a jogar. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

3.90  
Número Médio

**Senti-me "preso"** num nível, durante o jogo. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

3.10  
Número Médio

O jogo não é intrusivo e é fácil de entender. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

4.80  
Número Médio

O som e os gráficos do jogo **não** me ajudaram a mergulhar no ambiente. (0 ponto)

[Mais Detalhes](#)

[Informações](#)

10  
Respostas

1.70  
Número Médio

Consegui iniciar o jogo, sem a necessidade de ver o manual desde o menu inicial até sair do jogo.

[Mais Detalhes](#)

[Informações](#)

10

Respostas

4.30

Número Médio

0. Não consegui ter controle sobre as minhas ações e usa estratégias, sendo que no final os esforços não foram justos.

[Mais Detalhes](#)

[Informações](#)

10

Respostas

1.30

Número Médio

1. Achei que o conteúdo do jogo é apropriado, não contendo quaisquer conteúdos que incentive a violência ou ofensivo em termos de gênero, raça, religião e diversidade cultural.

[Mais Detalhes](#)

[Informações](#)

9

Respostas

5.00

Número Médio

2. A informação disponibilizada no jogo, **não** permitiu autoaprendizagem, **nem** que os conhecimentos sejam utilizados para outros contextos ou até mesmo no meu dia-a-dia.

[Mais Detalhes](#)

[Informações](#)

10

Respostas

1.50

Número Médio